FEATURE EXTRACTION AND PATTERN RECOGNITION FOR REAL-TIME EEG PROCESSING

BY

TEERA ACHARIYAPAOPAN

A DISSERTATION PRESENTED TO THE GRADUATE COUNCIL
OF THE UNIVERSITY OF FLORIDA
IN PARTIAL FULFILLMENT FOR THE REQUIREMENTS
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA
1983

ACKNOWLEDGMENTS

The author wishes to express his gratitude and esteem to his advisor, Dr. D. G. Childers, for his direction and support during this study.

He is also thankful to Dr. N. W. Perry, Jr., and Dr. I. S. Fischler for many stimulating discussions. The author wishes to thank Dr. M. A. Uman, Dr. T. E. Bullock and Dr. L. W. Couch for serving on the committee and for their guidance. He also wishes to express his appreciation to fellow graduate students, Dr. A. A. Arroyo and P. A. Bloom, for their stimulating discussions and encouragement.

He also wishes to thank David Glicksberg for his assistance in data collection.

He is thankful to his family and friends, Duangporn, Narane, Jo and John for their encouragement, moral support and their help in many ways.

TABLE OF CONTENTS

Abstract of Dissertation Presented to the Graduate Council
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

FEATURE EXTRACTION AND PATTERN RECOGNITION FOR REAL-TIME EEG PROCESSING

By

Teera Achariyapaopan

December 1983

Chairman: D. G. Childers
Major Department: Electrical Engineering

The application of pattern recognition techniques for classifying single EEG (Electroencephalographic) records is considered. This technique allows the processing of EEG signals in real-time. However, a problem arises in applying this technique. There are many potential features of the EEG signal and only a finite, usually small, number of training samples are available. It is well known that when a finite number of training samples are available for designing a classifier, there exists an optimal number of features which can be used to represent the data. This has led to the investigation of the effects of a finite number of training samples on feature extraction algorithms. We show that for a two-class, Gaussian data set with a common covariance matrix for each class, that additional features can be added to the feature vector one at a time if the value of an expression, which is a

function of the total number of training samples, number of features, and accumulated Mahalanobis distance, increases with each feature. For a fixed number of training samples, an equal training sample size for each class is suggested. An algorithm for determining the optimal number of features and the associated features is proposed. This algorithm is then applied to select features from EEG records, each of which consists of single evoked responses elicited from human subjects who read a series of statements. The subjects learned the truthfulness or falseness of each statement. The EEG records collected for the true and false statements make up our two class data base. A classifier is designed to assign these evoked event-related potentials (ERP's) to one class or the other (true or false) based on the features selected to represent the ERP's for each class. An "optimal" feature selection algorithm is described and compared with other feature selection techniques.

CHAPTER 1
INTRODUCTION

The electroencephalographic (EEG) signal measured from the scalp is a fluctuating electrical potential that reflects neuronal activity in the underlying brain structures. Most of the signal energy is below 50 Hz. The amplitude is typically 5 to 50 μv. Event-related potentials (ERP's) are a perturbation of the on-going EEG elicited by a single application of a sensory stimulus. The ERP's take place after some delay following the stimulus and last for about a second. The magnitude of the ERP's within the EEG are very small, typically 1-20 μv. Since the ERP amplitude is very small, it is difficult to detect the ERP signal buried in the on-going EEG, which is considered as noise in this case. ERP's are usually detected by averaging the EEGs for repeated stimulus presentation. But for some experiments, this detection problem is aggravated when only a limited number of replications of the evoking stimulus are available. In the past, ERP analysis has been done almost exclusively by signal averaging [1-3]. Clearly, if the ERP signal is deterministic, then averaging would be a good way to detect the signal. However, the ERP can be considered deterministic only as a rough approximation. Thus, there exists a need for more sophisticated techniques. In recent years, statistical pattern recognition and discriminant analysis techniques have been applied to process single

ERP's [4-12]. This technique enables one to analyze ERP's in real-time, which is useful for on-line experiments.

Pattern recognition is the process of classifying a measurement data set into mutually exclusive classes based on rules derived from a previously obtained training data set. Some authors [13-16] also include cluster analysis, a closely related subject, as part of pattern recognition. As in any pattern recognition system, the system for classifying ERP's can be considered as composed of two parts. These parts are feature extractor and classifier as shown in Figure 1.1. Usually the number of dimensions for the measurement vector $\underline{x}$ is large. This is especially true for the ERP signal. A large dimensional vector will complicate the decision algorithm. Moreover, it is well known [14, 17-20] that when a finite number of training samples is available for designing a classifier, there exists an optimal number of features for describing the data. Hence, there is a need to reduce the number of dimensions of the measurement vector while trying to retain as much class discriminatory ability as possible. The relationship between training sample size and the optimal number of features to be used has been studied [21-28]. Roucos and Childers [27-28] have given an analytical result of the relationship for these quantities for Gaussian data with known equal covariance matrices for each of two classes. Jain and Waller [24], El-Sheikh and Wacker [25], and Raudys and Pikelis [26] gave a similar result but for the unknown covariance matrix case. However, their formulae are very complicated and are difficult to apply. The contribution of this study is to derive a simpler

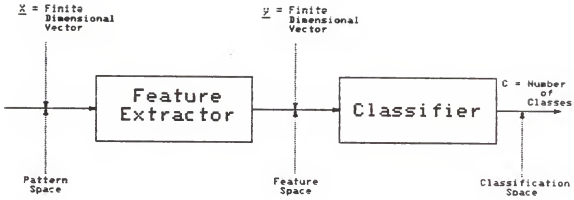


Figure 1.1 Conceptual recognition system.

expression. We also emphasize the importance of feature ordering and propose an algorithm for selecting features.

The second block in Figure 1.1 is the classifier. The classifier or decision rule partitions the feature space into c classes. Since it has been applied to many areas, for example, communications [29,30] and statistics [31-35], its theory is well established. From Figure 1.1, one might consider each block as independent, but actually each block is dependent. For example, one feature extractor method may be optimal only for one type of classifier. However, little is known about how these blocks interact.

The next three chapters discuss the design of the classifier. Chapter 2 considers the Bayesian classifier which depends on a complete knowledge of the underlying probabilities. In most practical situations, such a complete knowledge is not available and parameters have to be estimated from the training samples. This is the topic of Chapter 3. In Chapter 4, a different approach for designing a classifier is considered. This approach assumes a form for the classifier first and then estimates parameters of the classifier from the training samples. Only a linear model of this type of classifier is discussed. Chapter 5 reviews some of the existing feature selection and extraction techniques. The performance evaluation of a classifier is discussed in Chapter 6. Chapter 7 discusses the effects of a finite sample size on the performance of Gaussian classifiers. The optimal number of features and a feature selection algorithm are given in this chapter. These techniques are then applied to ERP data in Chapter 8.

Concluding remarks are provided in Chapter 9. Note that the pattern recognition theory discussed here is far from complete. This subject can be found in a number of books [13-15, 36-47]. Only the techniques believed to be useful for processing ERP's are presented in this dissertation.

CHAPTER 2
BAYESIAN DECISION THEORY

Bayesian decision theory is a fundamental statistical approach to the classification problem, which is based on the assumption that the decision problem is posed in probabilistic terms with a complete knowledge of all relevant probabilities. In this chapter we discuss the fundamentals of this theory and study a special case where the distributions are normal.

2.1 Bayes' Rule for Minimum Risk

Suppose that there are c possible pattern classes, ω_1, ω_2, ---, ω_c and an arbitrary pattern belongs to class ω_i with a priori probability P_i, $P_i > 0$ and $\sum_{i=1}^{c} P_i = 1$. The patterns are defined to be d-component measurement vectors or feature vectors. Thus an arbitrary pattern can be represented by a d-dimensional vector, $\underline{x}$. Pattern $\underline{x}$ is assumed to be a random vector assuming a value in d-dimensional feature space. The vector is described by a multivariate probability density function $p(\underline{x}|\omega_i)$, where pattern $\underline{x}$ is known to belong to class ω_i, i=1, ---, c. Let $\hat{\omega}(\underline{x})$ be some decision rule, i.e., a function of $\underline{x}$ that tells us which decision to make for every possible pattern $\underline{x}$. For example, $\hat{\omega}(\underline{x}) = \omega_i$ denotes the decision to assign pattern $\underline{x}$ to class ω_i. Let $\lambda(\omega_i|\omega_j)$ be the loss incurred when the decision $\hat{\omega}(\underline{x}) = \omega_i$ is made, when in fact the pattern belongs to ω_j, i=1, ---, c, j=1, ---, c.

Consider the problem of classifying an arbitrary pattern $\underline{x}$ of unknown class. We must decide to which class $\underline{x}$ belongs. The probability of $\underline{x}$ belonging to class ω_j is the class a posteriori probability $P(\omega_j|\underline{x})$. This probability can be calculated by Bayes' rule

$$P(\omega_j|\underline{x}) = \frac{p(\underline{x}|\omega_j)P_j}{p(\underline{x})} \tag{2.1}$$

where

$$p(\underline{x}) = \sum_{j=1}^{c} p(\underline{x}|\omega_j)P_j \tag{2.2}$$

is the unconditional probability function of $\underline{x}$. Now let us consider the loss associated with a particular decision. If we assign $\underline{x}$ to class i, i.e. $\hat{\omega}(\underline{x}) = \omega_i$, we will incur the loss of $\lambda(\omega_i|\omega_j)$ with probability $P(\omega_j|\underline{x})$. Consequently, the expected loss or conditional risk of making a decision $\hat{\omega}(\underline{x}) = \omega_i$ is

$$l^i(\underline{x}) = \sum_{j=1}^{c} \lambda(\omega_i|\omega_j)P(\omega_j|\underline{x}) \quad . \tag{2.3}$$

Note that the decision rule $\hat{\omega}(\underline{x})$ is a function which tells us which decision to make for every possible pattern $\underline{x}$. Then the conditional risk associated with this decision rule, $\hat{\omega}(\underline{x})$, is

$$l(\underline{x}) = \sum_{j=1}^{c} \lambda(\hat{\omega}(\underline{x})|\omega_j)P(\omega_j|\underline{x}) \tag{2.4}$$

and the average risk is given by

$$L = \overline{l(x)} = \int l(\underline{x})p(\underline{x})d\underline{x} \tag{2.5}$$

where the integral extends over the entire feature space.

Our problem is to find a decision rule $\hat{\omega}(\underline{x})$ which minimizes the average risk in (2.5). Clearly, this goal can be achieved by taking $\hat{\omega}(\underline{x})$ so that $l(\underline{x})$ in (2.4) is as small as possible for every $\underline{x}$. In other words Bayes' decision rule $\hat{\omega}(\underline{x})$ can be written as

$$\hat{\omega}^*(\underline{x}) = \omega_i \text{ if } l^i(\underline{x}) < l^j(\underline{x}),\ j=1, \text{---}, c \quad . \tag{2.6}$$

In case of a tie, any convenient tie-breaking rule can be used. The associated minimum conditional risk is

$$l^*(\underline{x}) = \min_{i=1, \text{---},c} l^i(\underline{x})$$

$$= \min_{i=1,\text{---},c} \sum_{j=1}^{c} \lambda(\omega_i|\omega_j)P(\omega_j|\underline{x}) \tag{2.7}$$

and the associated minimum average risk or Bayes' risk is

$$L^* = \int l^*(\underline{x})p(\underline{x})d\underline{x} \quad . \tag{2.8}$$

Note that under these conditions, no other decision rule can yield a smaller risk.

2.2 Bayes' Rule for Minimum Error Rate

So far our discussion has remained very general. In this section we consider a particular loss function. In many problems errors need to be avoided and all errors are equally costly. Thus a minimum error rate decision rule may be desired. A loss function associated with this type of problem has the form

$$\lambda(\omega_i|\omega_j) = 0 \quad \text{if} \quad i = j \tag{2.9}$$
$$= 1 \quad \text{if} \quad i \neq j \quad i,j = 1, ---,c \quad .$$

This loss function assigns no loss to a correct decision and assigns a unit loss to any error. With this loss function the conditional risk $l^i(\underline{x})$ of (2.3) is precisely the conditional probability of classification error associated with the decision $\hat{\omega}(\underline{x}) = \omega_i$, and is equal to

$$l^i(\underline{x}) = e^i(\underline{x}) = \sum_{\substack{j=1 \\ j\neq i}}^{c} P(\omega_j|\underline{x}) \tag{2.10}$$
$$= 1 - P(\omega_i|\underline{x}), \; i=1, ---,c$$

Bayes' decision rule in (2.6) becomes

$$\hat{\omega}^*(\underline{x}) = \omega_i \text{ if } e^i(\underline{x}) < e^j(\underline{x}), \; j = 1, ---,c \quad . \tag{2.11}$$

Decision rule (2.11) can be written in a more familiar form using the a posteriori probabilities, giving

$$\hat{\omega}^*(\underline{x}) = \omega_i \text{ if } P(\omega_i|\underline{x}) = \max_{j=1,\ ---,c} P(\omega_j|\underline{x}) \quad . \tag{2.12}$$

Note that from (2.12), to minimize the error rate, Bayes' rule suggests selecting i to maximize the a posteriori probability $P(\omega_i|\underline{x})$. The associated conditional probability of classification error and error rate are

$$\begin{aligned} e^*(\underline{x}) &= \min_{i=1,---,c} e^i(\underline{x}) \\ &= 1 - \max_{i=1,\ ---,c} P(\omega_i|\underline{x}) \end{aligned} \tag{2.13}$$

$$E^* = \int e^*(\underline{x}) p(\underline{x}) d\underline{x} \tag{2.14}$$

where the integral extends over the entire feature space.

2.3 Error Probabilities and Integrals

Bayes' decision rule (2.12) can be considered as a device for partitioning the feature space into c regions R_i, i=1, ---, c. The region R_i consists of those samples which are classified into class ω_i

$$R_i = \{\underline{x} | P(\omega_i|\underline{x}) = \max_{j=1, \ ---, c} P(\omega_j|\underline{x})\} \quad . \tag{2.15}$$

Thus the error rate (2.14) can be written as

$$E^* = \sum_{i=1}^{c} E^{*i} \tag{2.16}$$

where

$$E^{*i} = \int_{R_i} e^*(\underline{x}) p(\underline{x}) d\underline{x} \tag{2.17}$$

E^{*i} is the probability of error associated with the decision $\hat{\omega}^*(\underline{x}) = \omega_i$.

To obtain additional insight, let us consider the two-class case. Bayes' decision rule partitions the feature space into two regions

$$R_1 = \{\underline{x} | P(\omega_1|\underline{x}) > P(\omega_2|\underline{x})\} \tag{2.18a}$$

$$R_2 = \{\underline{x} | P(\omega_2|\underline{x}) > P(\omega_1|\underline{x})\} \quad . \tag{2.18b}$$

These regions are illustrated in Figure 2.1 for a one dimensional case. The error rate in this case is

$$E^* = E^{*1} + E^{*2}$$

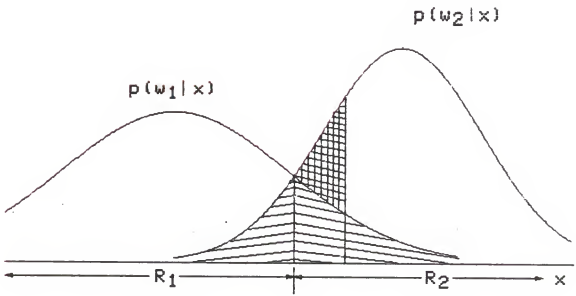


Figure 2.1 Components of the probability of error.

$$= \int_{R_1} (1 - P(\omega_1|\underline{x}))p(\underline{x})d\underline{x} + \int_{R_2} (1 - P(\omega_2|\underline{x}))p(\underline{x})d\underline{x}$$

$$= \int_{R_1} P(\omega_2|\underline{x})p(\underline{x})d\underline{x} + \int_{R_2} P(\omega_1|\underline{x})p(\underline{x})d\underline{x}$$

$$= P(\omega_2) \int_{R_1} p(\underline{x}|\omega_2)d\underline{x} + P(\omega_1) \int_{R_2} p(\underline{x}|\omega_1)d\underline{x} \qquad (2.19a)$$

$$E^* = P(x\epsilon R_1, \omega_2) + P(x\epsilon R_2, \omega_1) \quad . \qquad (2.19b)$$

E^* is the entire shaded area in Figure 2.1. Note that if the decision line is drawn elsewhere, the error rate is always greater than E^*. This shows that Bayes' decision rule yields the lowest error rate.

2.4 The Two-Class Case with Multivariate Normal Distributions

Our discussion so far has dealt with arbitrary probability densities. In this section we study the important case of normal distributions. We restrict ourselves to the two-class case. The normal or Gaussian probability density function is important because of its computational simplicity and because it represents a realistic model for many pattern classification situations.

When there are only two pattern classes, the decision rule (2.12) becomes

$$\hat{\omega}^*(\underline{x}) = \omega_1 \text{ if } P(\omega_1|\underline{x}) > P(\omega_2|\underline{x})$$

$$= \omega_2 \text{ otherwise} \tag{2.20}$$

Alternatively, the decision rule of (2.20) can be written in terms of the likelihood ratio $p(\underline{x}|\omega_1)/p(\underline{x}|\omega_2)$

$$\hat{\omega}^*(\underline{x}) = \omega_1 \text{ if } \frac{p(\underline{x}|\omega_1)}{p(\underline{x}|\omega_2)} > \frac{P(\omega_2)}{P(\omega_1)} \tag{2.21}$$

$$= \omega_2 \text{ otherwise.}$$

Now let us assume that each class is normally distributed with means $\underline{\mu}_i$ and covariance matrices $\sum_i$, i.e., $p(\underline{x}|\omega_i) \sim N(\underline{\mu}_i, \sum_i)$. More specifically,

$$p(\underline{x}|\omega_i) = (2\pi)^{-p/2} |\sum_i|^{-1/2} \exp[-1/2(\underline{x} - \underline{\mu}_i)^t \sum_i^{-1} (\underline{x} - \mu_i)] \tag{2.22}$$

where $(\underline{x} - \underline{\mu}_i)^t$ is the transpose of $(\underline{x} - \underline{\mu}_i)$, $\sum_i^{-1}$ is the inverse of $\sum_i$, $|\sum_i|$ is the determinant of $\sum_i$, and p is the dimension of $\underline{x}$. With this density, the likelihood ratio $p(\underline{x}|\omega_1)/p(\underline{x}|\omega_2)$ becomes

$$\frac{p(\underline{x}|\omega_1)}{p(\underline{x}|\omega_2)} = \frac{|\sum_1|^{-1/2} \exp[-\frac{1}{2}(\underline{x} - \underline{\mu}_1)^t \sum_1^{-1} (\underline{x} - \underline{\mu}_1)]}{|\sum_2|^{-1/2} \exp[-\frac{1}{2}(\underline{x} - \underline{\mu}_2)^t \sum_2^{-1} (\underline{x} - \underline{\mu}_2)]} \quad . \tag{2.23}$$

Since the logarithm is a monotonically increasing function, we can also consider the logarithm of the likelihood ratio $D(\underline{x})$ which is

$$D(\underline{x}) = \log \frac{p(\underline{x}|\omega_1)}{p(\underline{x}|\omega_2)}$$

$$= -\frac{1}{2}(\underline{x} - \underline{\mu}_1)^t \sum_1^{-1} (\underline{x} - \underline{\mu}_1) + \frac{1}{2}(\underline{x} - \underline{\mu}_2)^t \sum_2^{-1} (\underline{x} - \underline{\mu}_2)$$

$$+ \log \frac{|\sum_1|^{-1/2}}{|\sum_2|^{-1/2}} \quad . \tag{2.24}$$

Consequently, the decision rule (2.21) can be written as

$$\hat{\omega}^*(\underline{x}) = \omega_1 \text{ if } D(\underline{x}) > \log P(\omega_2) - \log P(\omega_1)$$

$$= \omega_2 \text{ otherwise} \tag{2.25}$$

$D(\underline{x})$ is sometimes known as a discriminant function. Now let us examine three special cases.

Case 1. $\sum_i = \sigma^2 I$

This simple case occurs when all features are statistically independent and have the same variance. Geometrically, this corresponds to the situation in which the samples fall into two equal size hyperspherical clusters, with the cluster for class ω_i being centered about the mean vector $\underline{\mu}_i$. Substituting $\sum^{-1} = 1/\sigma^2 I$ in (2.24), we obtain

$$D(\underline{x}) = -\frac{1}{2\sigma^2}\left[\|\underline{x} - \underline{\mu}_1\|^2 - \|\underline{x} - \underline{\mu}_2\|^2\right] \tag{2.26}$$

where

$$\|\underline{x} - \underline{\mu}_1\|^2 = (\underline{x} - \underline{\mu}_1)^t(\underline{x} - \underline{\mu}_1) \quad .$$

$\|\cdot\|$ denotes the Euclidean distance.

If the a priori probabilities $P(\omega_i)$ are the same for both classes, then $\log[P(\omega_2)/P(\omega_1)] = 0$ and the constant $1/2\sigma^2$ in (2.26) becomes unimportant and can be ignored. In this case the decision rule in (2.25) simply becomes

$$\hat{\omega}^*(x) = \omega_1 \text{ if } D(\underline{x}) = [\|x - \underline{\mu}_2\|^2 - \|x - \underline{\mu}_1\|^2] > 0$$

$$= \omega_2 \text{ otherwise.} \tag{2.27}$$

The classifier that uses the decision rule in (2.27) is known as a minimum distance classifier. The means for each class are thought of as an ideal prototype or template. An unknown pattern $\underline{x}$ is decided as belonging to class ω_i if the Euclidean distance between $\underline{x}$ and $\underline{\mu}_i$ is smallest of all such distances. This is essentially a template matching procedure.

If the a priori probabilities are not equal, then the decision rule in (2.25) becomes

$$\hat{\omega}^*(\underline{x}) = \omega_1 \text{ if } D(\underline{x}) > 0$$

$$= \omega_2 \text{ otherwise} \qquad (2.28a)$$

where

$$D(\underline{x}) = [\frac{1}{2\sigma^2}\|\underline{x} - \underline{\mu}_2\|^2 - \log P(\omega_2)] - [\frac{1}{2\sigma^2}\|\underline{x} - \underline{\mu}_1\|^2 - \log P(\omega_1)] \ . \qquad (2.28b)$$

Note that the distance is normalized by the variance σ^2 and is reduced $\log P(\omega_i)$. This can be considered a generalized minimum distance classifier.

By noting that $\underline{x}^t\underline{\mu} = \underline{\mu}^t\underline{x}$, the discriminant function (2.28b) can be written as

$$D(\underline{x}) = \frac{1}{2\sigma^2}[2\underline{x}^t(\underline{\mu}_1 - \underline{\mu}_2) - \underline{\mu}_1^t\underline{\mu}_1 + \underline{\mu}_2^t\underline{\mu}_2] - \log P(\omega_2) + \log P(\omega_1) \qquad (2.29)$$

which is a linear discriminant function. The classifier that uses such a linear discriminant function is called a linear machine.

Case 2. $\sum_i = \sum$

In this case both classes have an identical covariance matrix. Geometrically, this corresponds to the situation in which the samples

fall in hyperellipsoidal clusters of equal size and shape, the cluster for the class ω_i being centered about the mean vector $\underline{\mu}_i$. The decision rule for this case is

$$\hat{\omega}^*(\underline{x}) = \omega_1 \text{ if } D(\underline{x}) > 0$$

$$= \omega_2 \text{ otherwise} \qquad (2.30a)$$

where

$$D(\underline{x}) = [\frac{1}{2}(\underline{x}-\underline{\mu}_2)^t \sum_2^{-1}(\underline{x} - \underline{\mu}_2) - \log P(\omega_2)] -$$

$$[\frac{1}{2} (\underline{x} - \underline{\mu}_1)^t \sum_1^{-1}(\underline{x} - \underline{\mu}_1) - \log P(\omega_1)] \quad . \qquad (2.30b)$$

Note the similarity between (2.30) and (2.28). Again $D(\underline{x})$ in (2.30) can be considered as a generalized minimum distance discriminant function where in this case the distance is $(\underline{x} - \underline{\mu}_i)^t \sum^{-1}(\underline{x} - \underline{\mu}_i)$ which is known as the Mahalanobis distance. The discriminant function in (2.30) can be written as a linear function of $\underline{x}$ similar to (2.29).

$$D(\underline{x}) = \underline{x}^t \sum^{-1}(\underline{\mu}_1 - \underline{\mu}_2) - \frac{1}{2} (\underline{\mu}_1 + \underline{\mu}_2)^t \sum^{-1}(\underline{\mu}_1 - \underline{\mu}_2)$$

$$+ \log P(\omega_1) - \log P (\omega_2) \qquad (2.31)$$

when the a priori probabilities $P(\omega_i)$ are identical for both classes, the last two terms in (2.31) cancel out, and $D(\underline{x})$ becomes

$$D(\underline{x}) = (\underline{x}^t - \frac{1}{2}(\underline{\mu}_1 + \underline{\mu}_2)^t) \sum^{-1} (\underline{\mu}_1 - \underline{\mu}_2) \tag{2.32}$$

which is known as the Anderson linear discriminant plane. We will see in a later chapter that this discriminant function is also equivalent to the Fisher linear discrimination function.

Case 3. $\sum_i$ arbitrary.

The decision rule for this general case is (2.24) and (2.25). It is quadratic in nature and cannot be reduced. The decision surfaces are hyperquadrics, and can assume any of the general forms - pairs of hyperplanes, hyperspheres, hyperellipsoids, hyperparaboloids, or hyperhyperboloids of various types.

CHAPTER 3
PARAMETER ESTIMATION

In Chapter 2 we considered the problem of designing an optimal classifier where the a priori probabilities $P(\omega_i)$ and the class-conditional densities $p(\underline{x}|\omega_i)$ were completely known. Unfortunately, in pattern recognition applications this is usually not the case. In a typical situation we may have some vague, general knowledge about the problem together with a number of design samples. Then the problem is to find some way to use this information to design a classifier.

One approach to this problem is to use the training (design) samples to estimate the unknown probabilities and probability densities. The resulting estimates are used as if they were the true values. In a typical situation, the estimation of the a priori probabilities presents no serious problem. However, estimation of the class-conditional densities is not so simple. The number of training samples always seems too small and a serious problem arises when the dimensionality of the feature vector $\underline{x}$ is large. In this chapter we discuss how to estimate these class-conditional densities. We postpone the discussion of the effects of finite training sample size until Chapter 7.

For the estimation procedures discussed here, we assume that the general knowledge about the problem permits us to assume a form for the

density $p(\cdot\ ;\ \underline{\theta})$ except that it contains unknown parameters $\underline{\theta}$ (if $\underline{\theta}$ were known, the density function would be completely specified). This problem is known in statistics as parametric point estimation [48]. Other methods exist, which do not require knowledge about the form of density. These methods are known as nonparametric methods [14, 49, 50]. In general, the nonparametric methods require a large number of training samples. For the problem of classifying EEG data, a large number of training samples are usually not available. Thus, nonparametric estimation will not be treated here.

The parametric point estimation problem can be approached in several ways. We shall consider in the next two sections two common procedures, maximum likelihood estimation and Bayesian estimation respectively. To simplify our treatment, we shall assume that training samples from class ω_i give no information about $\underline{\theta}_j$ if $i \neq j$. That is equivalent to assuming that the parameters for the different classes are functionally independent. This permits us to work with each class separately and to simplify our notation by not having to indicate the class distinction. Thus we now have c separate problems for the c-class problem. Each separate problem is the following form. Use a set of samples, $\times = \{\underline{x}, \text{---}, \underline{x}_n\}$ drawn independently according to the probability law $p(\underline{x}\ ;\ \underline{\theta})$ to estimate the unknown parameter vector $\underline{\theta}$. $\underline{\theta}$ is written explicitly to indicate the dependence of the density $p(\underline{x})$ on the parameter vector $\underline{\theta}$.

3.1 Maximum Likelihood Estimation

Maximum likelihood methods view the parameters as quantities whose values are fixed but unknown. The best estimate is defined to be the one that maximizes the probability of obtaining the samples actually observed. Suppose that $\times$ contains n samples, $\times = \{\underline{x}_1, ---, \underline{x}_n\}$. Since the samples were drawn independently, then

$$p(\times\ ;\ \underline{\theta}) = \prod_{k=1}^{n} p(\underline{x}_k\ ;\ \underline{\theta}) \tag{3.1}$$

$p(\times\ ;\ \underline{\theta})$ is called the likelihood of $\underline{\theta}$ with respect to the set of samples. The maximum likelihood estimate of $\underline{\theta}$ is, by definition, the value $\hat{\theta}$ that maximizes $p(\times\ ;\ \underline{\theta})$. In some sense, this value corresponds to the value of $\underline{\theta}$ that best agrees with the actually observed samples.

Since the logarithm is monotonically increasing, the $\hat{\underline{\theta}}$ that maximizes the log-likelihood also maximizes the likelihood. Also, it is usually easier to work with the logarithm of the likelihood than with the likelihood itself. Let $l(\underline{\theta})$ be the log-likelihood function, then

$$l(\underline{\theta}) = \log p(\times\ ;\ \underline{\theta}) \quad . \tag{3.2}$$

Substitute (3.1) in (3.2),then

$$l(\underline{\theta}) = \sum_{k=1}^{n} \log p(\underline{x}_k\ ;\ \underline{\theta}) \quad . \tag{3.3}$$

If $l(\underline{\theta})$ is a differentiable function of $\underline{\theta}$, $\hat{\underline{\theta}}$ can be found by

differentiate $l(\underline{\theta})$ with respect to $\underline{\theta}$ and equating this derivative to $\underline{0}$. Let $\underline{\theta}$ be the p-component vector $\underline{\theta} = (\theta_1, ---, \theta_p)^t$ and let $\nabla_{\underline{\theta}}$ be the gradient operator, then with

$$\nabla_{\underline{\theta}} = [\frac{\partial}{\partial\theta_1}, ---, \frac{\partial}{\partial\theta_p}]^t \tag{3.4}$$

we have

$$\nabla_{\underline{\theta}} l(\theta) = \sum_{k=1}^{n} \nabla_{\underline{\theta}} \log p(\underline{x}_k ; \underline{\theta}) \quad . \tag{3.5}$$

$$= \underline{0}$$

The vector $\hat{\underline{\theta}}$ can be found by solving (3.5).

As an example, suppose that the samples $x = \{x_1, ---, x_n\}$ are drawn from a normal population with mean μ and variance σ^2 (Consider the univariate case for simplicity). That is denoted as $x \sim N(\mu, \sigma^2)$. We know that x is normally distributed and samples $x = \{x_1, ---, x_n\}$ are available. The problem is to find the maximum likelihood estimate of μ and σ^2. (If we know the parameters μ and σ^2, then, of course, $p(x)$ is completely known). Let $\theta_1 = \mu$ and $\theta_2 = \sigma^2$. Here

$$\log p(x_k ; \underline{\theta}) = -\frac{1}{2} \log 2\pi\theta_2 - \frac{1}{2\theta_2} (x_k - \theta_1)^2$$

and

$$\nabla_{\underline{\theta}} \log p(x_k ; \underline{\theta}) = \begin{bmatrix} \frac{1}{\theta_2} (x_k - \theta_1) \\ -\frac{1}{2\theta_1} + \frac{(x_k - \theta_1)^2}{2\theta_2^2} \end{bmatrix} .$$

Then, Eq. (3.5) leads to the conditions

$$\sum_{k=1}^{n} \frac{1}{\hat{\theta}_2} (x_k - \hat{\theta}_1) = 0$$

and

$$- \sum_{k=1}^{n} \frac{1}{\hat{\theta}_2} + \sum_{k=1}^{n} \frac{(x_k - \hat{\theta}_1)^2}{\hat{\theta}_2} = 0$$

where $\hat{\theta}_1$ and $\hat{\theta}_2$ are the maximum likelihood estimates for θ_1 and θ_2 respectively. By substituting $\hat{\mu} = \hat{\theta}_1$, $\hat{\sigma}^2 = \hat{\theta}_2$ and rearranging, we obtain the maximum likelihood estimates for μ and σ^2; denoted as

$$\hat{\mu} = \frac{1}{n} \sum_{k=1}^{n} x_k \tag{3.6}$$

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{k=1}^{n} (x_k - \hat{\mu})^2 \quad . \tag{3.7}$$

Similarly for the multivariate case, one can show that the maximum likelihood estimates for $\underline{\mu}$ and Σ are given by

$$\hat{\underline{\mu}} = \frac{1}{n} \sum_{k=1}^{n} \underline{x}_k \tag{3.8}$$

and

$$\hat{\Sigma} = \frac{1}{n} \sum_{k=1}^{n} (\underline{x}_k - \hat{\underline{\mu}})(\underline{x}_k - \hat{\underline{\mu}})^t \quad . \tag{3.9}$$

Note that the maximum likelihood estimate for the mean vector is the sample mean. The maximum likelihood estimate for the covariance matrix is the average of the n matrices $(\underline{x}_k - \hat{\underline{\mu}})(\underline{x}_k - \hat{\underline{\mu}})^t$. This is a quite satisfying result since the true covariance matrix is the expected value of the matrix $(\underline{x} - \underline{\mu})(\underline{x} - \underline{\mu})^t$. However, it is well known that the maximum likelihood estimate for a covariance matrix is biased; that is, the expected value of $\hat{\sum}$ is not equal to $\sum$. An unbiased estimate for $\sum$ is given by the sample covariance matrix

$$S = \frac{1}{n-1} \sum_{k=1}^{n} (\underline{x}_k - \hat{\underline{\mu}})(\underline{x}_k - \hat{\underline{\mu}})^t \quad . \tag{3.10}$$

3.2 Bayesian Estimation

The basic assumptions for Bayesian estimation are:

(1) The form of the density $p(\underline{x}|\underline{\theta})$ is assumed to be known, but the value of the parameter vector $\underline{\theta}$ is not known.

(2) The parameter vector $\underline{\theta}$ is assumed to be a random variable. Our initial knowledge about $\underline{\theta}$ is assumed to be contained in a known a priori density $p(\underline{\theta})$.

(3) The rest of our knowledge about $\underline{\theta}$ is contained in a set $\times$ of n samples $\underline{x}_1$, ---, $\underline{x}_n$ drawn independently according to the unknown probability law $p(\underline{x})$.

The problem is to compute $p(\underline{x}|\times)$, yielding the best approximation of $p(\underline{x})$. Note that the difference between the Bayesian method and the maximum likelihood method is the second assumption. The Bayesian method

views the parameter vector $\underline{\theta}$ as a random variable having a known a priori density $p(\underline{\theta})$. The maximum likelihood method views the parameter vector $\underline{\theta}$ as a quantity whose value is fixed but unknown.

To obtain $p(\underline{x}|\times)$ note that

$$p(\underline{x}|\times) = \int p(\underline{x},\underline{\theta}|\times)d\underline{\theta} \tag{3.11}$$

where the integral extends over the entire parameter space. Now we can write

$$p(\underline{x},\underline{\theta}|\times) = p(\underline{x}|\underline{\theta},\times)p(\underline{\theta}|\times) \quad . \tag{3.12}$$

Since $\underline{x}$ and $\times$ are independent, that is the distribution of $\underline{x}$ is completely known once we know the value of $\underline{\theta}$. Then the first factor in the right side of (3.12) is merely $p(\underline{x}|\underline{\theta})$. Thus from (3.11) and (3.12)

$$p(\underline{x}|\times) = \int p(\underline{x}|\underline{\theta})\ p(\underline{\theta}|\times)d\underline{\theta} \quad . \tag{3.13}$$

This equation links the desired density $p(\underline{x}|\times)$ to the a posteriori density $p(\underline{\theta}|\times)$ for the unknown parameter vector. The only unknown in (3.13) is the a posteriori density $p(\underline{\theta}|\times)$. To calculate $p(\underline{\theta}|\times)$, by Bayes' rule

$$p(\underline{\theta}|\times) = \frac{p(\times|\underline{\theta})p(\underline{\theta})}{\int p(\times|\underline{\theta})p(\underline{\theta})d\underline{\theta}} \tag{3.14}$$

and by the independence assumption

$$p(\underline{x}|\underline{\theta}) = \prod_{k=1}^{n} p(\underline{x}_k|\underline{\theta}) \quad . \tag{3.15}$$

This constitutes the formal solution to the problem. We show its relation to the maximum likelihood solution. Suppose that $p(\underline{x}|\underline{\theta})$ has a sharp peak at $\underline{\theta} = \hat{\underline{\theta}}$. If the a priori density $p(\underline{\theta})$ is not zero at $\underline{\theta} = \hat{\underline{\theta}}$ and does not change much in the surrounding neighborhood, then $p(\underline{\theta}|\underline{x})$ also peaks at $\underline{\theta} = \hat{\underline{\theta}}$. Thus (3.13) shows that $p(\underline{x}|\underline{x})$ is approximately $p(\underline{x}|\hat{\underline{\theta}})$, a result that one would obtain using the maximum likelihood estimate as if it were the true value.

To illustrate the method, let us consider the normal case. For simplicity, we will consider the univariate case. Thus the density is $p(x|\mu) \sim N(\mu,\sigma^2)$. Suppose that the mean μ is the only unknown parameter (σ^2 is known). We assume that whatever prior knowledge we might have about μ can be expressed by a known a priori density $p(\mu)$ and we shall further assume that

$$p(\mu) \sim N(\mu_o,\sigma_o^2) \tag{3.16}$$

where both μ_o and σ_o^2 are known. Generally speaking, μ_o is our best initial guess for μ, and σ_o^2 is the measure of the uncertainty of this guess. The assumption that the a priori distribution for μ is normal will simplify the problem. However, the important point is not that the a priori distribution for μ is normal, but that it exists and is

known. Suppose now that n samples x_1, ---, x_n are independently drawn from the resulting population. Let $\times = \{x_1, ---, x_n\}$, by Bayes' rule in (3.14), we obtain

$$p(\mu|\times) = \frac{p(\times|\mu)p(\mu)}{\int p(\times|\mu)p(\mu)d\mu}$$

$$= \alpha \prod_{k=1}^{n} p(x_k|\mu)p(\mu) \tag{3.17}$$

where α is a scale factor that depends on $\times$ but is independent of μ. Equation (3.17) shows how the observation of a set of samples affect our ideas about the true value of μ, changing the a priori density $p(\mu)$ into an a posteriori density $p(\mu|\times)$. Since $p(x_k|\mu) \sim N(\mu,\sigma^2)$ and $p(\mu) \sim N(\mu_o,\sigma_o^2)$,

$$p(\mu|\times) = \alpha \prod_{k=1}^{n} \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[-\frac{1}{2}\left(\frac{x_k-\mu}{\sigma}\right)^2\right] \frac{1}{\sqrt{2\pi}\,\sigma_o} \exp\left[-\frac{1}{2}\left(\frac{\mu-\mu_o}{\sigma_o}\right)^2\right]$$

$$= \alpha' \exp\left[-\frac{1}{2}\left\{\sum_{k=1}^{n}\left(\frac{\mu-x_k}{\sigma}\right)^2 + \left(\frac{\mu-\mu_o}{\sigma_o}\right)^2\right\}\right]$$

$$= \alpha'' \exp\left[-\frac{1}{2}\left\{\left(\frac{n}{\sigma^2}+\frac{1}{\sigma_o^2}\right)\mu^2 - 2\left(\frac{1}{\sigma^2}\sum_{k=1}^{n} x_k + \frac{\mu_o}{\sigma_o^2}\right)\mu\right\}\right] \tag{3.18}$$

where α' and α'' are factors that do not depend on μ. $p(\mu|\times)$ in (3.18) is still a normal distribution for any number for samples, n, and $p(\mu|\times)$ is said to be a reproducing density. We can write

$p(\mu|x) \sim N(\mu_n, \sigma_n^2)$ where

$$\mu_n = \frac{n\sigma_o^2}{n\sigma_o^2 + \sigma^2} m_n + \frac{\sigma^2}{n\sigma_o^2 + \sigma^2} \mu_o \tag{3.19}$$

$$\sigma_n^2 = \frac{\sigma_o^2 \sigma^2}{n\sigma_o^2 + \sigma^2} \tag{3.20}$$

where m_n is the sample mean

$$m_n = \frac{1}{n} \sum_{k=1}^{n} x_k \quad . \tag{3.21}$$

Roughly speaking, μ_n represents our best guess for μ after observing n samples, and σ_n^2 measures our uncertainty about this guess. Since σ_n^2 decreases monotonically with n, each additional observation decreases our uncertainty about the estimate of μ. As n increases, $p(\mu|x)$ becomes more and more sharply peaked, approaching an impulse function as n approaches infinity. This behavior is generally known as Bayesian learning (see Fig. 3.1).

It is obvious from (3.19) that μ_n is a linear combination of the sample mean m_n and the initial guess μ_o, with coefficients that are nonnegative and sum to one. Thus, μ_n always lies somewhere between m_n and μ_o. If $\sigma_n \neq 0$, μ_n approaches the sample mean m_n as n approaches infinity. If $\sigma_o = 0$, we have a degenerate case in which our a priori certainty that $\mu = \mu_o$ is so strong that no number of observations can change our opinion. At the other extreme, if $\sigma_o >> \sigma$, we are so

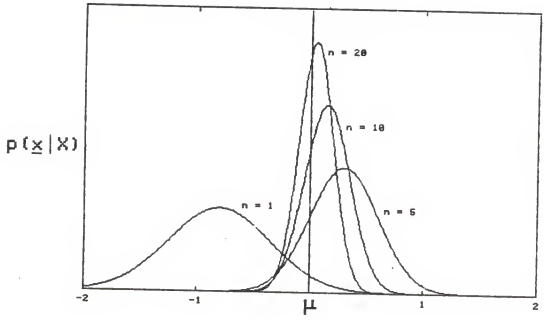


Figure 3.1 Learning the mean of a normal density.

uncertain about our a priori guess that we take $\mu_n = m_n$, using only the samples to estimate μ.

Having found the a posteriori density $p(\mu|\mathcal{X})$, we must next find the density $p(x|\mathcal{X})$. From (3.13),

$$p(x|\mathcal{X}) = \int p(x|\mu)p(\mu|\mathcal{X})d\mu$$

$$= \int \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right] \frac{1}{\sqrt{2\pi}\,\sigma_n} \exp\left[-\frac{1}{2}\left(\frac{\mu-\mu_n}{\sigma_n}\right)^2\right]d\mu$$

$$= \frac{1}{2\pi\sigma\sigma_n} \exp\left[-\frac{1}{2}\frac{(x-\mu_n)^2}{\sigma^2+\sigma_n^2}\right] f(\sigma,\sigma_n)$$

where

$$f(\sigma,\sigma_n) = \int \exp\left[-\frac{1}{2}\frac{\sigma^2+\sigma_n^2}{\sigma^2\sigma_n^2}\left(\mu - \frac{\sigma_n^2 x+\sigma^2\mu_n}{\sigma^2+\sigma_n^2}\right)^2\right] d\mu \quad .$$

Again, $p(x|\mathcal{X})$ is normally distributed with mean μ_n and variance $\sigma^2+\sigma_n^2$

$$p(x|\mathcal{X}) \sim N(\mu_n, \sigma_n^2 + \sigma^2) \quad . \tag{3.22}$$

In summary, the density $p(x|\mathcal{X})$ whose parametric form is known to be $p(x|\mu) \sim N(\mu,\sigma^2)$, we merely replace μ by μ_n and σ^2 by $\sigma^2 + \sigma_n^2$. In effect, the conditional mean μ_n is treated as if it were the true mean, and the known variance is increased to account for the additional uncertainty in x resulting from our lack of exact knowledge about the mean μ. From the density $p(x|\mathcal{X})$, which is actually the class-

conditional density $p(x|\omega_i, x_i)$, and together with the a priori probability $P(\omega_i)$ we obtain the needed information to design the Bayes classifier.

CHAPTER 4
LINEAR DISCRIMINANT FUNCTIONS

In Chapter 2 we assumed that the forms for the underlying probability distributions were known and used training samples to estimate the values of the distributions parameters. This method may lead to a complicated discriminant function which could be difficult to implement. In this chapter, we take a different approach by assuming the form of discriminant function, and then use the training samples to estimate the values of its parameters. We will examine various procedures for determining these parameters. These procedures have been regarded as the learning method. None of these procedures require knowledge of the underlying distributions, consequently we can select any form of the discriminant function we desire. One of the most popular discriminant functions is the linear function, because this function can be implemented easily and is optimal in some cases if the underlying distributions are of the proper form. Throughout this chapter we will be concerned with the linear discriminant function only. We will also limit ourselves to the two-class case. The multiclass case is a straightforward extention of the two-class case and can be found in a number of texts on pattern recognition [14,37,42,45,47].

4.1 Linear Discriminant Function for the Two-Class Case

A linear discriminant function has a form

$$g(\underline{x}) = w_1x_1 + w_2x_2 + \text{---} + w_px_p + w_o$$
$$= \underline{w}^t\underline{x} + w_o \quad . \tag{4.1}$$

The p-dimensional vector $\underline{w} = (w_1, w_2, \text{---}, w_p)^t$ is called the weight vector and w_o is the threshold weight. The decision rule corresponding to the discriminant function $g(\underline{x})$ is

$$\hat{\omega}(\underline{x}) = \omega_1 \quad \text{if } g(\underline{x}) > 0$$
$$= \omega_2 \quad \text{otherwise} \quad . \tag{4.2}$$

If $g(\underline{x}) = 0$, an arbitrary class can be assigned. The decision rule partitions the feature space into two regions, one where $g(\underline{x}) > 0$ and another where $g(\underline{x}) < 0$. The decision boundary is determined by $g(\underline{x}) = 0$. Since $g(\underline{x})$ is a linear function in $\underline{x}$, the decision surface is a p-dimensional hyperplane and is given by

$$\underline{w}^t\underline{x} = -w_o \quad . \tag{4.3}$$

Let us review some geometric properties of the hyperplane. Let $\underline{x}_1$ and $\underline{x}_2$ both lie on the hyperplane, then we have

$$\underline{w}^t \underline{x}_1 + w_o = \underline{w}^t \underline{x}_2 + w_o = 0 \quad .$$

Thus

$$\underline{w}^t (\underline{x}_1 - \underline{x}_2) = 0 \quad . \tag{4.4}$$

Since the vector $(\underline{x}_1 - \underline{x}_2)$ lies in the hyperplane, (4.4) shows that the vector $\underline{w}^t$ is normal to the decision surface and thus defines the orientation of the hyperplane. The normal Euclidean distance from the origin to the hyperplane is

$$\frac{\underline{w}^t \underline{x}_1}{\|\underline{w}\|} = -\frac{w_o}{\|\underline{w}\|} \quad . \tag{4.5}$$

With a proper normalization, the threshold weight w_o defines the location of the hyperplane. Let r be the signed, normal Euclidean distance from an arbitrary point $\underline{x}'$ to the hyperplane, where r is positive if $g(\underline{x}) > 0$ and negative if $g(\underline{x}) < 0$. Then r is equal to (see Figure 4.1)

$$r = \frac{\underline{w}^t}{\|\underline{w}\|} (\underline{x}' - \underline{x}_1) = \frac{\underline{w}^t \underline{x}' + w_o}{\|\underline{w}\|} = \frac{g(\underline{x}')}{\|\underline{w}\|} \quad . \tag{4.6}$$

Thus the distance r is proportional to $g(\underline{x})$.

In short, the linear discriminant function partitions the feature space by the hyperplane into two regions. The orientation of the hyperplane is determined by the normal vector $\underline{w}$ and the location of the

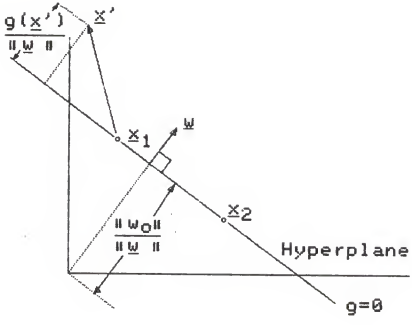


Figure 4.1 The geometry of the hyperplane $g(\underline{x}) = 0$.

hyperplane is determined by the threshold weight w_o. In particular, if the threshold weight $w_o = 0$, the hyperplane passes through the origin. The origin is on the positive side (i.e., when $g(\underline{x}) > 0$) if $w_o > 0$, and on the negative side when $g(\underline{x}) < 0$. The discriminant function $g(\underline{x})$ is proportional to the signed distance from $\underline{x}$ to the hyperplane.

Before we consider methods for determining the weights of the discriminant function, let us introduce a convention for our notation. Let $\underline{y}$ denote the (p+1) dimensional vector

$$\begin{aligned} \underline{y} &= [x_1, x_2 \text{ ---}, x_p, 1]^t \\ &= [\underline{x}, 1]^t \quad . \end{aligned} \tag{4.7}$$

Let $\underline{a}$ denote the (p+1) dimensional weight vector

$$\begin{aligned} \underline{a} &= [w_1, w_2, \text{---}, w_p, w_o]^t \\ &= [\underline{w}, w_o]^t \quad . \end{aligned} \tag{4.8}$$

With this notation, (4.1) can be written as

$$g(\underline{x}) = \underline{a}^t \underline{y} \quad . \tag{4.9}$$

The vector $\underline{y}$ is known as the augmented pattern vector.

4.2 Deterministic Learning Algorithm

Now we shall investigate some nonstatistical methods for determining the weight vector $\underline{a}$. The first method, originated by Rosenblatt [51] is known as the perceptron learning algorithm. This algorithm assumes that the training samples are linearly separable. The second algorithm to be discussed is known as the least mean-squared-error procedure. The training samples for the second procedure can not necessarily be linearly separated. The training samples are said to be linearly separable if there exists a linear classifier that can classify them correctly. In other words there exists a vector $\underline{a}$ such that

$$\underline{a}^t \underline{y}_1 > 0 \quad \forall \underline{y}_1 \in \omega_1 \tag{4.10a}$$

$$\underline{a}^t \underline{y}_1 < 0 \quad \forall \underline{y}_1 \in \omega_2 \quad . \tag{4.10b}$$

To simplify this treatment, let us introduce a useful convention. Note that (4.10b) can be written as $\underline{a}^t(-\underline{y}_1) > 0, \forall \underline{y}_1 \in \omega_2$. This suggests a normalization procedure achieved by replacing all the samples from class ω_2 by their negative values. With this normalization, we can forget the labels assigned to the training samples and look for a weight vector $\underline{a}$ such that $\underline{a}^t \underline{y}_1 > 0$. This convention will be used throughout this section.

Other deterministic learning algorithms exist than the two we shall discuss, e.g., the potential function approach, the gradient technique, etc. [14, 45, 47]. However, we will not discuss these here.

4.2.1. The Perceptron Learning Algorithm

Suppose that we have a set of n samples $S_n = \{\underline{y}_1, ---, \underline{y}_n\}$, some labeled ω_1 and some labeled ω_2 and S_n is normalized, i.e. $\underline{y}_i$ is replaced by $-\underline{y}_i$ if $\underline{y}_i \in \omega_2$. Assume that these training samples are linearly separable, i.e., there exist a weight vector $\underline{a}$ such that

$$g(\underline{y}_i) = \underline{a}^t \underline{y}_i > 0 \quad \forall \underline{y}_i \in S_n \quad . \tag{4.11}$$

Our problem is to find such a vector. This problem is essentially one of solving a set of linear inequalities, which can be done in many ways. One such method is an iterative procedure, which we now describe. Let $\underline{a}(k)$ denote the weight vector at kth iterative step of the algorithm, and $\underline{y}(k)$ denote the kth sample at kth step obtained by scanning the sample set S_n repeatedly and cyclically, i.e. $\underline{y}(k) = \underline{y}_i$ if $i = k \bmod n.$, $k = 1, 2, ---$. The algorithm can be written as

$$\underline{a}(k+1) = \underline{a}(k) + c(k)\underline{y}(k) \quad \text{if } \underline{a}^t(k)\underline{y}(k) < 0 \tag{4.12a}$$

$$= \underline{a}(k) \qquad \text{otherwise} \tag{4.12b}$$

where c(k) is the positive correction factor at kth step.

The idea of this algorithm is to leave the weight $\underline{a}$ unchanged if it classifies the sample correctly and move the weight vector in the "right" direction when it makes a mistake. The solution is obtained

when $\underline{a}$ correctly classifies all n samples. Note that the solution is not unique and can be any vector in the solution region (see Figure 4.2). The initial weight can be assumed arbitrarily. To gain some insight into how the algorithm works, let us examine the kth step. Assume $\underline{y}(k)$ belongs to ω_1. If $\underline{a}^t(k)\underline{y}(k) > 0$, the weight vector $\underline{a}$ is left unchanged, otherwise $\underline{a}(k+1) = \underline{a}(k) + c(k)\underline{y}(k)$. Note that

$$\underline{a}^t(k+1)\underline{y}(k) = \underline{a}(k)^t\underline{y}(k) + c(k)\underline{y}^t(k)\underline{y}(k) > \underline{a}^t(k)\underline{y}(k) \quad . \quad (4.13)$$

Thus the weight adjustment is trying to correct the error, which it may or may not correct. This is controlled by the correction factor c. If c(k) is independent of k, the procedure is called the fixed-increment rule. If c(k) is the smallest integer that makes $\underline{a}^t(k+1)\underline{y}(k) > 0$, i.e. the smallest integer greater than $|\underline{a}^t(k)\underline{y}(k)|/|\underline{y}^t(k)\underline{y}(k)|$, the procedure is called the absolute correction rule. Another possibility is to take c(k) as a fraction of $|\underline{a}^t(k)\underline{y}(k)|/|\underline{y}^t(k)\underline{y}(k)|$, this procedure is known as the fractional correction rule. It should be noted that if the correction factor is too small, the algorithm will take longer to converge. But if the correction factor is too large, the algorithm may overcorrect. The proof of the covergence of the algorithm can be found in [14,45,47,52].

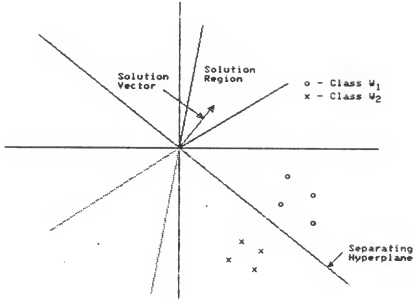


Figure 4.2 Linearly separable samples and the solution region in weight space.

4.2.2 The Least Mean-Squared-Error Procedure

The procedure in the last subsection depends on the assumption that training samples are linearly separable. If this is not the case, the algorithm will never converge. Unfortunately, sufficiently large experimental design sets are almost certainly not linearly separable. We shall now consider an algorithm which does not require linearly separable training samples. This algorithm pays attention to all the training samples, whereas the last procedure focussed only on the misclassified samples. Previously we have sought a weight vector $\underline{a}$, making the inner products $\underline{a}^t\underline{y}_i > 0$. In this section, we are trying to make $\underline{a}^t\underline{y}_i = b_i$ where the b_i are some arbitrarily specified positive constants. Thus, we have replaced the problem of solving a set of linear inequalities with a problem of solving a set of linear equations.

To ease the discussion, let us introduce a matrix notation. Let Y be the n by (p+1) matrix whose ith row is the vector $\underline{y}_i^t$, and $\underline{b}$ be the column vector $(b_1, \text{---}, b_n)^t$. Then our problem is to find a weight vector $\underline{a}$ satisfying

$$\underline{Ya} = \underline{b} \quad . \tag{4.14}$$

If Y is a square nonsingular matrix, we would obtain $\underline{a} = Y^{-1}\underline{b}$. However, Y is usually a rectangular matrix with more rows than columns or, in other words, there are more equations than unknowns. The vector $\underline{a}$ is overdetermined and generally there is no solution. However, we can seek

a weight vector $\underline{a}$ that minimizes the sum of squares of the error $\underline{e} = Y\underline{a} - \underline{b}$. This is equivalent to minimizing

$$\nabla J(\underline{a}) = \|Y\underline{a} - \underline{b}\|^2 = \sum_{i=1}^{n} (\underline{a}^t \underline{y}_i - b_i)^2 \quad . \tag{4.15}$$

A solution can be found by forming the gradient of $J(\underline{a})$ and setting it to zero, i.e.,

$$\nabla J(\underline{a}) = \sum_{i=1}^{n} 2(\underline{a}^t \underline{y}_i - b_i)\underline{y}_i$$

$$= 2Y^t(Y\underline{a} - \underline{b}) = 0 \tag{4.16}$$

This yields the condition

$$Y^tY\underline{a} = Y^t\underline{b}. \tag{4.17}$$

Y^tY is usually a nonsingular square matrix. Then $\underline{a}$ can be obtained by

$$\underline{a} = (Y^tY)^{-1}Y^t\underline{b}$$

$$= Y^{\#}\underline{b} \tag{4.18}$$

where

$$Y^{\#} = (Y^tY)^{-1}Y^t$$

is called the pseudoinverse [53].

It is clear that the solution vector $\underline{a}$ depends on the margin vector $\underline{b}$. Different values of $\underline{b}$ give different solution vectors $\underline{a}$ with different properties. Thus, for an arbitrary value of vector $\underline{b}$, there is no reason to believe that the LMSE solution yields a separating vector in the linear separable case. However, it is reasonable to expect that the solution obtained gives a reasonable result in both separable and nonseparable cases. To support this expectation, Koford and Groner [54] have shown that with a proper value of vector $\underline{b}$, the LMSE discriminant function $\underline{a}^t\underline{y}$ is directly related to Fisher's linear discriminant which will be discussed in the next section. Patterson and Womack [55] showed that the LMSE solution algo gave a minimum-squared-error approximation to the Bayes' discriminant. There is a variation of this procedure known as Ho-Kashyap algorithm [56] which gives a separating vector for the linear separable case.

4.3 Fisher's Linear Discriminant

The linear discriminant function in this section is rather different from those in the previous section. Fisher's linear discriminant does not solve the classification problem. Its purpose is to reduce the dimensionality of the data from p-dimensions to one dimension and still retain class separability. More specifically, we want to find a linear transformation that maps the data from feature space into one dimension and this mapping should maximize some criterion functions. This criterion function serves as the indication of class separability.

Suppose that we have a set of n p-dimensional samples $X = \{\underline{x}_1, \text{---}, \underline{x}_n\}$ in which n_1 samples come from class ω_1 and $n_2 = n-n_1$ samples come from ω_2. If we form a linear combination of the components of $\underline{x}$, we obtain a scalar

$$z = \underline{w}^t \underline{x} \tag{4.19}$$

for each of the n samples, i.e., $Z = \{z_1, \text{---}, z_n\}$. Geometrically, if $\|\underline{w}\| = 1$, each z_i is the projection of the corresponding x_i onto a line in the direction of $\underline{w}$. This is illustrated in Figure 4.3. Actually, the magnitude of $\underline{w}$ is not important. It is merely a scaling factor. The important thing is the direction of $\underline{w}$. Figure 4.3 also illustrates that in one direction, namely $\underline{w}$, we obtain a good separation of the classes, while in another direction, $\underline{w}'$, the data is completely mixed.

To obtain a good separation, we would like the projected points from the same class to cluster together, but the two clusters should be separated. Thus we would like the projected points to have a large difference of the sample means and have small within group variances. If $\underline{m}_i$ is the p-dimensional sample mean given by

$$\underline{m}_i = \frac{1}{n_i} \sum_{\underline{x} \in X_i} \underline{x} \tag{4.20}$$

then the sample mean for the projected points is given by

$$\tilde{m}_i = \frac{1}{n_i} \sum_{z \in Z_i} z$$

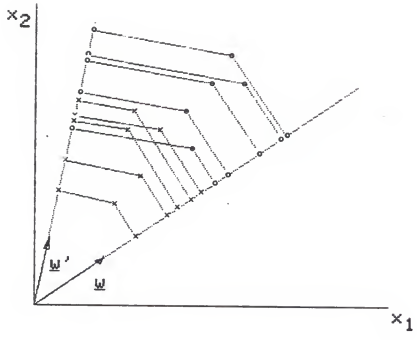


Figure 4.3 Projection of samples onto a line.

$$= \frac{1}{n_i} \sum_{\underline{x} \in X_i} \underline{w}^t \underline{x} = \underline{w}^t \underline{m}_i \tag{4.21}$$

It follows that the difference of the sample means of the projected points is given by

$$|\tilde{m}_1 - \tilde{m}_2| = |\underline{w}^t(\underline{m}_1 - \underline{m}_2)| \tag{4.22}$$

Define the scatter of the projected samples for class ω_i as

$$\tilde{S}_i^2 = \sum_{z \in Z_i} (z - \tilde{m}_i)^2, \quad i = 1,2 \tag{4.23}$$

Thus, an estimate of the variance of the pooled data is $(1/n)$ $(\tilde{S}_1^2 + \tilde{S}_2^2)$. Then, rather than using the variance of the pooled data, we can use $(\tilde{S}_1^2 + \tilde{S}_2^2)$ which is known as the within-class scatter of the projected samples. The Fisher linear discriminant [57] is then defined as the linear function $\underline{w}^t\underline{x}$ for which the criterion function

$$J(\underline{w}) = \frac{|\tilde{m}_1 - \tilde{m}_2|^2}{\tilde{S}_1^2 + \tilde{S}_2^2} \tag{4.24}$$

is maximized.

To obtain J as an explicit function of $\underline{w}$, we can rewrite $\tilde{S}_i^2$ in (4.23) as

$$\tilde{S}_i^2 = \sum_{\underline{x} \in X_i} (\underline{w}^t \underline{x} - \underline{w}^t \underline{m}_i)^2$$

$$= \sum_{x \in X_i} \underline{w}^t(\underline{x} - \underline{m}_i)(\underline{x} - \underline{m}_i)^t\underline{w}$$

$$\tilde{s}_i^2 = \underline{w}^t S_i \underline{w} \tag{4.25}$$

where

$$S_i = \sum_{x \in X_i} (\underline{x} - \underline{m}_i)(\underline{x} - \underline{m}_i)^t \tag{4.26}$$

is called the scatter matrix. Then the within-class scatter equals to

$$\tilde{s}_1^2 + \tilde{s}_2^2 = \underline{w}^t(S_1 + S_2)\underline{w}$$

$$= \underline{w}^t S_w \underline{w} \tag{4.27}$$

where

$$S_w = S_1 + S_2 \tag{4.28}$$

is called the within-class scatter matrix. The within-class scatter matrix is proportional to the sample covariance matrix for the pooled p-dimensional data, is symmetric and positive semidefinite, and is usually nonsingular if $n > p$. Similarly, $|\tilde{m}_1 - \tilde{m}_2|^2$ can be written as

$$(\tilde{m}_1 - \tilde{m}_2)^2 = (\underline{w}^t\underline{m}_1 - \underline{w}^t\underline{m}_2)^2$$
$$= \underline{w}^t(\underline{m}_1 - \underline{m}_2)(\underline{m}_1 - \underline{m}_2)^t\underline{w} \tag{4.29}$$
$$= \underline{w}^t S_B \underline{w}$$

where

$$S_B = (\underline{m}_1 - \underline{m}_2)(\underline{m}_1 - \underline{m}_2)^t \tag{4.30}$$

is called the between-class scatter matrix, which is also symmetric and positive semidefinite, but because $(\underline{m}_1 - \underline{m}_2)$ is a vector, i.e. pxl matrix, its rank is at most one [58].

In terms of S_B and S_w, the criterion function J can be written as

$$J(\underline{w}) = \frac{\underline{w}^t S_B \underline{w}}{\underline{w}^t S_w \underline{w}} \quad . \tag{4.31}$$

It can be shown that the $\underline{w}$ that maximizes J in (4.31) is [14,46]

$$\underline{w} = S_w^{-1} (\underline{m}_1 - \underline{m}_2) \tag{4.32}$$

which is Fisher's linear discriminant, i.e., the linear function which maximizes the ratio of the between-class scatter to within-class scatter. Thus, the p-dimensional problem has been converted to a one-dimensional problem. It will be seen in the next chapter that Fisher's linear discriminant can be used to reduce the dimensionality of the data, via feature extraction and feature selection. Actually, Fisher's linear discriminant can be extended to convert the data from p-dimensions to d-dimensions for $p \geq d \geq 1$ [59]. Note that this mapping can not theoretically reduce the minimum achievable error rate. In general, one has to sacrifice some of the theoretically attainable performance for the advantage of being able to work in a lower dimension. However, if the conditional density functions $p(\underline{x}|\omega_i)$ (i=1,2) are multivariate normal with equal covariance matrices, one need not sacrifice performance. Recall from Chapter 2 that for this case the discriminant function is (Eq. (2.31) in Chapter 2)

$$D(\underline{x}) = \underline{w}^t \underline{x} + w_o \tag{4.33}$$

where

$$\underline{w} = \sum^{-1} (\underline{\mu}_1 - \underline{\mu}_2) \tag{4.34}$$

$$w_o = -\frac{1}{2} (\underline{\mu}_1 + \underline{\mu}_2)^t \sum^{-1} (\underline{\mu}_1 - \underline{\mu}_2) + \log P(\omega_1) + \log P(\omega_2) \; . \tag{4.35}$$

If the estimates of the sample means and the sample covariance matrix, $\underline{\mu}_i$ and $\sum$ are used, the $\underline{w}$'s in (4.34) and (4.32) are the same. Thus the samples are mapped into the same direction except for Fisher's linear discriminant. The threshold weight w_o must be estimated separately.

It is also interesting to compare (4.19) and (4.1). These two equations are almost the same except (4.1) has an additive term, the threshold weight w_o. Thus the linear discriminant function in (4.1) can be interpreted in a manner similar to that above, i.e., each point $\underline{x}$ is projected onto a line in the $\underline{w}$ direction, then the origin of the new axis is translated to the right (in the positive direction) by w_o. Every point falling on the positive side of this new origin is decided as coming from class ω_1, otherwise ω_2. By the same token Fisher's linear discriminant can be interpreted as a method for finding the hyperplane which partitions the feature space. However Fisher's linear discriminant only gives the direction (orientation) $\underline{w}$ of the

hyperplane. The discriminant cannot determine the position, i.e. the threshold weight. This is why we stated earlier that Fisher's linear discriminant does not solve the classification problem.

CHAPTER 5
FEATURE SELECTION AND EXTRACTION

So far, we have only dealt with the problem of designing a classifier. In this chapter we discuss feature selection and extraction, whose purpose is to reduce the number of features required to represent the data while retaining the class discrimination information. The main motivation for reducing the number of features is to simplify the classifier and to avoid the dimensionality problem. The dimensionality problem is a phenomenon recently observed by many authors [14, 17-20]. This phenomenon may be described as follows. Consider the situation where only a finite number of training samples for two classes of data are available. The performance of a classifier for this training set is not necessarily improved as one increases the number of features representing the data. In fact, the classifier performance may even deteriorate as we add features to our feature set. This will be discussed more fully in Chapter 7.

There are two feature reduction techniques. One is feature selection in the measurement space, known as feature selection. The other is feature selection in the transform space and is known as feature extraction. Feature selection is simply an algorithm for choosing a subset of p measurements (features) that maximize some criterion function. Thus an optimal selection can be achieved only by testing all possible sets of p features chosen from P measurements,

i.e., by applying the criterion $\binom{P}{p} = p!/(p!(P - p)!)$ times . The difficulty with this method is that the number of calculations required is enormous even for a modest number of features. In practice, some suboptimal procedures, such as the search algorithm known as branch and bound, or heuristic approach is used. Some of these methods can be found in [47,60]. In Chapter 7 we suggest an algorithm for solving this problem.

For feature extraction, the problem is to find a p x P matrix W such that the derived features $\underline{y} = W\underline{x}$ maximize some criterion. All feature extraction algorithms can be classified into two categories. The first category solve the matrix W directly. This method is similar to Fisher's linear discriminant method described in Chapter 4. For the second category the data is transformed to another domain, for instance by the Fourier transform, the Hadamand transform, the Karhunen-Loève expansion, etc. Then features are selected from the transformed domain. In this chapter, we discuss several of these methods. In Section 5.2, we discuss a method suggested first by Foley and Sammon [59]. This method is an extension of Fisher's linear discriminant and falls in the first category. We then describe in Section 5.3 a feature extraction algorithm belonging to the second category. This method is based on Karhunen-Loève expansion. Since all feature extraction algorithms require a feature selection criterion, we will first discuss this topic in Section 5.1.

5.1 Feature Selection Criteria

The ultimate goal of a pattern recognition system is to properly classify the data with as low a classification error rate as possible. Thus an ideal criterion for feature selection is to minimize the error rate. Unfortunately, the error rate is difficult to evaluate. To overcome this theoretical problem other criteria are chosen which vary monotonically with the error rate. The functions are selected to be easily calculated. Generally, these criteria should possess the following properties.

(1) Should be monotonically increasing or decreasing with the probability or error, or with the bound (upper or lower) on probability of error.

(2) Should be invariant under one-to-one mapping. This property is important since the probability of error is invariant under any transformation which holds one-to-one correspondence.

(3) All the selected features should be uncorrelated.

(4) Satisfy the metric properties, i.e.

(i) $c(\omega_i, \omega_j : Y_l) > 0$ for $i \neq j$

(ii) $c(\omega_i, \omega_i : Y_l) = 0$

(iii) $c(\omega_i, \omega_j : Y_l) = c(\omega_j, \omega_i : Y_l)$

(iv) $c(\omega_i, \omega_j : Y_l) \leq c(\omega_i, \omega_j : Y_{l+1})$

where Y_l denotes a subset of l features, ω_i denotes class ω_i and c( ) denotes a criterion.

Note that all the above properties are not necessary but are desirable. So far for the general case, none of the existing criteria

satisfy all of the above conditions. Some of the criteria that have been suggested are divergence [61], Bayesian distance [62], Matusita distance [63], Bhattacharyya [63], etc. All of these measures are well documented in [47,64], so we will not discuss them here. The only criterion that we will be using in the rest of this chapter is Fisher's ratio.

5.2 Feature Extraction Based on Discriminant Analysis

For this type of feature extraction, the feature vectors are a linear combination of the measurement vectors which minimize or maximize a criterion and the features are mutually orthogonal. In other words, let J be a criterion and $\underline{x}$ be a P-dimensional measurement vector. We wish to find a p-dimensional vector $\underline{y}$ ($p < P$) such that

$$\underline{y} = W\underline{x} \tag{5.1}$$

where

$$W = [\underline{w}_1, \underline{w}_2, \cdots, \underline{w}_p]^t$$

$$= \text{an } p \times P \text{ matrix such that } \underline{w}_i \quad \underline{w}_j = 0 \text{ if } i \neq j$$

and $\underline{y}$ maximizes the criterion J. An example of this type of features extractor is the Foley-Sammon method [59]. Foley-Sammon suggest Fisher's ratio as a criterion, i.e.,

$$J = \frac{(\underline{w}^t \Delta)^2}{\underline{w}^t S_w \underline{w}} \tag{5.2}$$

where

$\underline{w}$ = P-dimensional column vector onto which the data are projected

$\underline{w}^t$ = transpose of $\underline{w}$

$\underline{x}_j^{(i)} = (x_{j1}^{(i)}, \ldots, x_{jp}^{(i)})$ jth sample vector for class ω_i

n_i = the number of sample in class ω_i

$\hat{\underline{\mu}}$ = estimated mean of class ω_i, $\hat{\underline{\mu}}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \underline{x}_j^{(i)}$

$\underline{\Delta}$ = difference in the estimated means, $\underline{\Delta} = \hat{\underline{\mu}}_1 - \hat{\underline{\mu}}_2$

S_i = within-class scatter for class ω_i, $S_i = \sum_{j=1}^{n_i} (\underline{x}_j^{(i)} - \hat{\underline{\mu}}_i)(\underline{x}_j^{(i)} - \hat{\underline{\mu}}_i)^t$

S_W = sum of the within-class scatter, $S_W = S_1 + S_2$.

From (4.32) in Section 4.3,

$$\underline{w}_1 = \alpha_1 S_W^{-1} \underline{\Delta} \tag{5.3}$$

where $\underline{w}_1$ is the direction of the best feature and α_1 is chosen such that $\underline{w}_1^t \underline{w}_1 = 1$.

The other $\underline{w}_i$'s for $i > 1$ are subjected to the following contraints, $\underline{w}_i$ maximize the criterion J and $\underline{w}_i$ is orthogonal to $\underline{w}_j$ for $j < i$. Foley and Sammon [59] give a sequential algorithm to achieve this.

5.3 Feature Extraction Based on the Karhunen-Loève Expansion

Before describing how the Karhunen-Loève expansion may be utilized for feature extraction, let us consider the expansion itself. Only the discrete-time case will be considered here. For the continuous case see [65].

Consider an ensemble of P-dimensional random vectors $\underline{x}$ and suppose that any sample vector $\underline{x} = [x_1, x_2, ---, x_p]^t$ from this ensemble belongs to one of the m possible pattern classes ω_i, i=1,---,m, where the probability of occurrence of the ith class is $P(\omega_i)$. Let us assume also that each class has been centralized by subtracting the means, μ_i of the random vector $\underline{x}_i$ in that class. Thus, if the centralized observation from the class ω_i is denoted by $\underline{z}_i$ we can write,

$$\underline{z}_i = \underline{x} - \underline{\mu}_i \quad .$$

Suppose now that we wish to represent $\underline{z}_i$ by a special finite expansion of the form,

$$z_i = \sum_{k=1}^{P} v_{ki} \underline{u}_k \tag{5.4}$$

where the $\underline{u}_k$ are orthonormal deterministic vectors satisfying the condition

$$\underline{u}^t_k \underline{u}^*_l = \delta_{kl} \tag{5.5}$$

while the v_{ki} are zero mean and mutually uncorrelated random coefficients for which

$$\sum_{i=1}^{m} P(\omega_i) E\{v_{ki} v^*_{li}\} = \rho_k^2 \delta_{kl} \tag{5.6}$$

with the * in both (5.5) and (5.6) denoting the complex conjugate, and δ_{kl} represents the Kronecker delta function, i.e.

$$\delta_{kl} = \begin{cases} 1 & , \quad k = l \\ 0 & , \quad k \neq l \end{cases} .$$

The deterministic vectors $\underline{u}_k$ in equation (5.4) are called the Karhunen-Loève coordinate vectors. Chien and Fu [66] have shown that these vectors are the eigenvectors of the covariance matrix C of $\underline{z}$, where

$$C = \sum_{i=1}^{m} P(\omega_i) E\{\underline{z}_i \underline{z}_i^{t\,*}\} \tag{5.7}$$

and that $\lambda_k = \rho_k^2$ are their associated eigenvalues.

The importance of the Karhunen-Loève (K-L) coordinate vectors $\underline{u}_k$ in signal processing becomes apparent if they are arranged in a descending order, according to the magnitude of their corresponding eigenvalues λ_k, i.e.,

$$\lambda_1 \geq \lambda_2 \text{ --- } \geq \lambda_p \geq \text{ --- } \geq \lambda_P \quad . \tag{5.8}$$

For then, if an approximation of $\hat{\underline{z}}_i$ of $\underline{z}_i$ is constructed of the form

$$\hat{\underline{z}}_i = \sum_{k=1}^{p} v_{ki}\, \underline{u}_k \tag{5.9}$$

where $p < P$, it can be shown [66] that both the mean square error $\bar{e}^2$,

$$\bar{e}^2 = \sum_{i=1}^{m} P(\omega_i)\, E\{|\underline{z}_i - \hat{\underline{z}}_i|^2\} \tag{5.10}$$

and the entropy function H_p, where

$$H_p = \sum_{k=p}^{P} \rho_k^2 \log \rho_k^2 \tag{5.11}$$

are minimized with respect to the ordering of the eigenvectors. Moreover, Tou and Heydorn [67] pointed out that the total entropy $H = -\sum_{k=1}^{P} \rho_k^2 \log \rho_k^2$ associated with the coordinated system $\underline{u}_k, k=1,---,p$, obtained in this manner is minimized with respect to any other coordinate system.

In order words, by ordering the eigenvectors $\underline{u}_k$ in correspondence with the decreasing magnitude of their associated eigenvalues, we obtain the optimal reduced K-L coordinate system in which the first p coordinate coefficients, v_{ki}, defined as

$$v_{ki} = \underline{z}_i^t \underline{u}_k \tag{5.12}$$

contain most of the "information for reconstruction" of the random vector $\underline{x}_i$.

Finally, it should be noted that if the transformation in equation (5.12) is applied to the original pattern vector $\underline{x}$ with means included, then we obtain a new feature vector $\underline{y}$. This transformation can be written in vector-matrix terms as

$$\underline{y}^t = \underline{x}^t U \tag{5.13}$$

where U is a matrix composed of the complete set of ordered eigenvectors, i.e.,

$$U = [\underline{u}_1, \underline{u}_2, \text{---}, \underline{u}_p] \quad . \tag{5.14}$$

The eigenvalues λ_k of the new covariance matrix are then the variances ρ_k^2 of these new features, i.e.

$$\lambda_k = \rho_k^2 = \sum_{i=1}^{m} P(\omega_i) \text{var}(y_{ki}) \tag{5.15}$$

where

$$\text{var}(y_{ki}) = E\{(y_{ki} - v_{ki})^2\} \tag{5.16}$$

and v_{ki} is the transformed mean of the feature y_{ki}, i.e.,

$$v_{ki} = \underline{\mu}_i^t \cdot \underline{u}_k \quad . \tag{5.17}$$

A number of feature extraction techniques based on the K-L expansion have been suggested in recent years. These techniques can be classified into six major categories.

(1) In the first approach suggested by Chien and Fu [66], the K-L expansion is based on coordinate vectors $\underline{u}_k = [u_{1k}, u_{2k}, \text{---}, u_{kk}]^t$

which are chosen as the eigenvectors of the covariance matrix C, where

$$C = \sum_{i=1}^{m} P(\omega_i)\, E\{[\underline{x}_i - \underline{\mu}_i][\underline{x}_i - \underline{\mu}_i]^t\} \quad . \tag{5.18}$$

As we have seen previously, if the eigenvectors are ordered according to the magnitude of their associated eigenvalues λ_k, then the mean square error $\bar{e}^2$ and the H_p as defined in (5.10) and (5.11), respectively, are both minimized by taking the first p features ($p < P$) of the transformed feature vector $\underline{y}$, where

$$\underline{y}^t = \underline{x}^t \quad U \quad . \tag{5.19}$$

(2) The second approach was suggested by Tou and Heydorn [67]. Here the transformation matrix $U_p(Pxp)$ is formed by the first p eigenvectors of the same covarience matrix C as in (5.18) above. But their choice of the eigenvectors is based on the smallest eigenvalue of C, i.e.

$$\lambda_1 < \lambda_2 < \text{---} < \lambda_p < \text{---} \lambda_P \quad . \tag{5.20}$$

Tou and Heydorn [67] show that the entropy $H_p = -\sum_{k=1}^{p} \rho_k^2 \log \rho_k^2$ associated with the p features formed in this way is minimized. This technique provides a very good description of each individual class but there is no guarantee that the selected features will have

any discriminatory power between several classes.

(3) A third approach, due to Watanabe et al. [68], is called Clafic uses a K-L expansion where the coordinate system is optimal with respect to one class ω_i alone. Here, the covariance matrix C_i is given by

$$C_i = E[\underline{x}\underline{x}^t] \quad \underline{x}\varepsilon\omega_i \quad . \tag{5.21}$$

In this case,the means of the representation vector $\underline{x}$ are not subtracted. If the transformation matrix U is composed of eigenvectors assoicated with the eigenvalues λ_k of the matrix C_i placed in descending order, then the information contained in the transformed feature vector $\underline{y}$, where $\underline{y}$ is defined by

$$\underline{y}^t = \underline{x}^t U \quad x\varepsilon\omega_i$$

is concentrated on the first few coordinates. But when input patterns $\underline{x} \not\varepsilon \omega_i$, the information in the corresponding feature vector $\underline{y}$ will be spread over all the components y_k. In this way a measure of the dispersion of information can be used for classification purposes.

(4) The fourth method is also due to Watanabe et al. [68], and is called Selfic. In this method, the covariance matrix C is computed from the input patterns $\underline{x}$ centralized over all classes by subtracting the mean vector $\underline{\mu}$ defined as

$$\underline{\mu} = E[\underline{x}] \quad \forall \underline{x} \quad . \tag{5.22}$$

In other words, the covariance matrix C is defined here as

$$C = E[(\underline{x} - \underline{\mu})(\underline{x} - \underline{\mu})^t] \quad \forall \underline{x} \quad . \tag{5.23}$$

Thus, the eigenvalues of C are functions of both the variances and the means of each class. The eigenvectors are arranged in descending order according to their corresponding eigenvalues. It is suggested that those features y_k with small eigenvalues will contain most of the information about the membership of the pattern $\underline{x}$ in the appropriate class.

(5) Fukunaga and Koontz have suggested their transform [69]. The patterns $\underline{x}$ are first transformed by a matrix Q such that the covariance matrix C of the resulting vectors is a unit matrix, i.e.

$$C = \sum_{i=1}^{2} P(\omega_i) E[Q^t \underline{x}_i \underline{x}_i^t Q] = I \quad . \tag{5.24}$$

Application of the K-L expansion to each class separately corresponds to finding the eigenvalues and eigenvectors of the matrices $C_i, i=1,2$, where C_i is defined as

$$C_i = P(\omega_i) E[Q^t \underline{x}\underline{x}^t Q] \quad x \varepsilon \omega_i \quad . \tag{5.25}$$

In this way, the sum C of these matrices C_i satisfies the condition

$$C = C_1 + C_2 = I \quad .$$

Fukunaga and Koontz [69] have shown that the two classes have the following properties:

(i) the two systems of eigenvectors are identical

(ii) the important features are never shared (the largest eigenvalue of one class is the smallest of the other).

As a result, the new basis vectors are formed by eigenvectors corresponding to the largest eigenvalues of each class.

(6) The last approach has been suggested by Roucos and Childers [28,70]. This method is similar to the fourth method, but instead of ordering the features by their eigenvalues, they are ordered according to Fisher's ratio.

CHAPTER 6
PERFORMANCE ESTIMATION

In the previous chapters we discussed the design of a classifier. After the classifier is designed, one needs to evaluate its performance to compare the design with competing designs. The error rate is the performance measure that will be discussed here.

The estimation of the error rate for a classifier has received considerable attention in the literature. An extensive bibliography has been published by Toussaint [71]. There are many approaches to the problem. In this chapter we discuss only a few of these approaches.

6.1 Empirical Approach

This approach counts the number of errors when testing the classifier on a test data set, which can be chosen in numerous ways. Here we describe four popular procedures.

(1) The Resubstitution Estimate. In this procedure the same data set is used for both designing and testing the classifier. Many authors [72-74] have shown both experimentally and theoretically that this procedure gives a very optimistic estimate, especially when the data set is small. Note, however, that when a large data set is available, this method is as good as any procedure.

(2) The Holdout Estimate. The data is partitioned into two mutually exclusive subsets in this procedure. One set is used for designing the classifier and the other for testing. This procedure makes poor use of the data since a classifier designed on the entire data set will, on the average, perform better than a classifier designed on only a portion of the data set. This procedure is known to give a very pessimistic error estimate.

(3) The Leave-One-Out Estimate. This procedure assumes that there are n data samples available. Remove one sample from the data set. Design the classifier with the remaining (n-1) data samples and then test it with the removed data sample. Return the sample removed earlier to the data set. Then repeat the above steps, removing a different sample each time, n times until every sample has been used for testing. The total number of errors is the leave-one-out error estimate. Clearly this method uses the data very effectively. This method is sometimes referred to as the Jack Knife method.

The leave-one-out estimate was shown experimentally by Lachenbruch and Mickey [73] to be approximately unbiased, a property independent of the type of the classifier or the underlying distribution of the data. However, there are at least two disadvantages of the method. Firstly, the unbiased property is achieved at the expense of an increase in the variance of the estimator. Secondly, it is clear that this procedure requires a great deal of computation. However, Fukunaga and Kessel [75] have

shown that for the multivariate Gaussian case the leave-one-out procedure requires little extra computation.

(4) The Rotation Estimate. In this procedure the data set is partitioned into n/d disjoint subsets, where d is a divisor of n. Then, remove one subset from the design set, design the classifier with the remaining data and test it on the removed subset, not used in the design. Repeat the operation for n/d times until every subset is used for testing. The rotation estimate is the average frequency of misclassification over the n/d test sessions.

When d = 1 the rotation method reduces to the leave-one-out method. When d = n/2 it reduces to the holdout method where the roles of the design and test sets are interchanged. The interchanging of design and test sets is known in statistics as cross-validation in both directions. As we may expect, the properties of the rotation estimate will fall somewhere between the leave-one-out method and holdout method. The rotation estimate will be less biased than the holdout method and the variance is less than the leave-one-out method.

6.2 Parametric Approach

The estimates in the last section do not assume a form for the distribution function or the type of the classifer. In this section, we will assume that the form of the distribution function is known but the parameters are not. These parameters have to be estimated from the available training samples. There are a number of possible error rates

which may be defined, and each one may be valuable depending on the circumstances.

Let $E(A_1,A_2)$ be the error rate which depends upon the classification regions, A_1, as defined in Section 2.3, and the presumed distribution, A_2, of the observations that will be classified. The arguments A_1 and A_2 are written explicitly to emphasize the fact that the error rate E depends on both arguments, i.e., the classification regions and the presumed distribution of the observations that will be classified. Note that specifying the classification regions is equivalent to specifying a specific classifier. Let f denote the presumed distribution of the observations to be classified. R denotes the classification regions when the parameters are known, i.e., f is known. In other words, R is the classification regions of the classifier which is designed with a complete knowledge of the underlying distributions, f. Let $\hat{f}$ denote the presumed distribution of the observations with its parameters estimated from N training samples in which n_1 samples come from ω_1 and n_2 samples from ω_2. Let $\hat{R}$ be the classification regions when the parameters are unknown, i.e., estimated from N training samples. In a sense $\hat{f}$ and $\hat{R}$ are the estimates of f and R, respectively. To avoid confusion between the expectation notation E, and error rate E, let E[ ] denote the expectation operator with square brackets. The error rates of interest are as follows.

(1) The optimum error rate. The optimum error rate is the error rate when all parameters are known. In our notation, the optimum error rate is represented by E(R,f). For the Bayes

classifier, the optimum error rate is the Bayes error rate E* defined in (2.18) and (2.19) of Chapter 2. As an example, let us consider a two-class case multivariate normal distribution with equal covariance matrix and a priori probability. From Section 2.4, the decision rule for this case is

$$\begin{aligned} \hat{\omega}^*(\underline{x}) &= \omega_1 \quad \text{if } D(\underline{x}) > 0 \\ &= \omega_2 \quad \text{otherwise} \end{aligned} \tag{6.1}$$

where

$$D(\underline{x}) = (\underline{x} - \frac{1}{2}(\underline{\mu}_1 + \underline{\mu}_2))^t \sum^{-1} (\underline{\mu}_1 - \underline{\mu}_2) \quad . \tag{6.2}$$

Thus the corresponding optimum classification regions are

$$\begin{aligned} R_1 &= \{\underline{x} | D(\underline{x}) > 0\} \\ R_2 &\quad \text{otherwise} \end{aligned} \quad . \tag{6.3}$$

Then the optimum error rate is given by

$$E(R,f) = \frac{1}{2} \int_{R_1} p(\underline{x}|\omega_2) d\underline{x} + \frac{1}{2} \int_{R_2} p(\underline{x}|\omega_1) d\underline{x}$$

$$= \frac{1}{2} Pr(D(\underline{x}) > 0 | \omega_2) + \frac{1}{2} Pr(D(\underline{x}) < 0 | \omega_1) \quad . \tag{6.4}$$

When all parameters are known the statistic of D is normally distributed with parameters

$$E[D(\underline{x})|\omega_i] = \frac{(-1)^{i+1}}{2}(\underline{\mu}_1 - \underline{\mu}_2)^t \sum^{-1}(\underline{\mu}_1 - \underline{\mu}_2)$$
$$= \frac{(-1)^{i+1}}{2}\delta^2 \qquad (6.5)$$

$$\mathrm{Var}(D(\underline{x})) = E[D(\underline{x}) - D(\underline{\mu}_i)]^2$$
$$= (\underline{\mu}_1 - \underline{\mu}_2)^t \sum^{-1} (\underline{\mu}_1 - \underline{\mu}_2) \qquad (6.6)$$
$$= \delta^2 \quad .$$

The quantity δ^2 is the Mahalanobis distance with known parameters. Then the optimum error rate is

$$E(R,f) = \phi(-\frac{\delta}{2}) \qquad (6.7)$$

where $\phi(z)$ is the standard normal distribution function defined as

$$\phi(z) = \frac{1}{2\pi}\int_{-\infty}^{z} \exp(-\frac{1}{2}u^2)\,du \quad . \qquad (6.8)$$

(2) The actual error rate. The actual error rate is the error rate of a classifier using estimated parameters. We denote the actual rate by $E(\hat{R},f)$. For a two-class case it is given by

$$E(\hat{R},f) = P_2 \int_{\hat{R}_1} p(\underline{x}|\omega_2)d\underline{x} + P_1 \int_{\hat{R}_2} p(\underline{x}|\omega_1)d\underline{x} \quad . \qquad (6.9)$$

Consider a two-class case multivariate normal distribution with equal covariance matrix and a priori probability. The decision rule is given by

$$\hat{\omega}(\underline{x}) = \omega_1 \text{ if } D_s(\underline{x}) > 0$$
$$= \omega_2 \text{ otherwise} \tag{6.10}$$

where

$$D_s(\underline{x}) = [\underline{x} - \frac{1}{2}(\underline{m}_1 + \underline{m}_2)]^t S^{-1}(\underline{m}_1 - \underline{m}_2) \tag{6.11}$$

where $\underline{m}_i$ and S are the sample mean of $\underline{\mu}_i$ and the pooled unbiased estimate of the covariance matrix, respectively i.e.,

$$\underline{m}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \underline{x}_j^{(i)} \tag{6.12}$$

$$S = \frac{1}{n_1 + n_2 - 2} \sum_{i=1}^{2} \sum_{j=1}^{n_i} (\underline{x}_j^{(i)} - \underline{m}_i)^2 \quad . \tag{6.13}$$

In the above expression $\underline{x}_j^{(i)}$ refers to the jth training sample from class ω_i. The corresponding classification regions are

$$R_1 = \{\underline{x} | D_s(\underline{x}) > 0\}$$
$$R_2 \quad \text{otherwise} \tag{6.14}$$

The actual error rate for this case is

$$E(\hat{R},f) = \frac{1}{2} \Pr(\underline{x}\epsilon R_1 | \omega_2) + \frac{1}{2} \Pr(\underline{x}\epsilon R_2 | \omega_1)$$

$$= \frac{1}{2} \Pr[D_s(\underline{x}) > 0 | \omega_2] + \frac{1}{2} \Pr[D_s(\underline{x}) < 0 | \omega_1] \quad (6.15)$$

$D_s(\underline{x})$ is a conditioned normal distribution with known $\underline{m}_1, \underline{m}_2$, and S. The means and variances are given by

$$E[D_s(\underline{x}) | \omega_i] = [\underline{\mu}_i - \frac{1}{2}(\underline{m}_1 + \underline{m}_2)]^t S^{-1}(\underline{m}_1 - \underline{m}_2)$$

$$= D_s(\underline{\mu}_i) \quad (6.16)$$

$$\text{Var } D_s(\underline{x}) = (\underline{m}_1 - \underline{m}_2)^t S^{-1} \sum S^{-1}(\underline{m}_1 - \underline{m}_2)$$

$$= V_D \quad . \quad (6.17)$$

The probability that $\underline{x}$ falls in R_1, if $\underline{x}$ belongs to ω_2, is

$$\Pr(D_s(\underline{x}) > 0 | \omega_2) = \Pr[\frac{D_s(\underline{x}) - D_s(\underline{\mu}_2)}{\sqrt{V_D}} > -\frac{D_s(\underline{\mu}_2)}{\sqrt{V_D}}]$$

$$= \phi[\frac{D_s(\underline{\mu}_2)}{\sqrt{V_D}}] \quad . \quad (6.18)$$

Similarly

$$\Pr(D_s(\underline{x}) < 0 | \omega_1) = \phi[-\frac{D_s(\underline{\mu}_1)}{\sqrt{V_D}}] \quad . \quad (6.19)$$

Thus

$$E(\hat{R},f) = \frac{1}{2} \phi[-\frac{D_s(\underline{\mu}_1)}{\sqrt{V_D}}] + \frac{1}{2} \phi[\frac{D_s(\underline{\mu}_2)}{\sqrt{V_D}}] \quad . \quad (6.20)$$

(3) The expected error rate. The expected error rate is the expected value of the actual error rate over all possible samples of size n_1, and n_2, which is sometimes known as the expected actual error rate. In other words, the expected error rate is the average of the actual error rate when n_1 and n_2 training samples from ω_1 and ω_2, respectively, are available. For a two-class case multivariate normal distribution with equal covariance matrix, the expected error rate is

$$E[E(\hat{R},f)] = P_2 \cdot \Pr[D_s(\underline{x}) > 0 | \omega_2] + P_1 \cdot \Pr[D(\underline{x}) < 0 | \omega_1] \ . \tag{6.21}$$

The distribution of D_s is very complex. Many statisticians have studied this problem [76-91]. John [76-78] gave the exact distribution and the associated expected error rate of D_s when the covariance matrix is known. The unknown covariance matrix case has been studied by Sitgreaves [79]. Approximations have been given by various authors [82-91].

Lachenbruch [92] suggested an estimate for the expected actual error rate based on the expected mean and variance of $D_s(\underline{x})$. The expected means and variance for the sample size n_1 and n_2 for class 1 and 2 are

$$E[D_s(\underline{x}) | \omega_i] = \frac{1}{2} C_1 [(-1)^{i+1} \delta^2 - \frac{p(n_2-n_1)}{n_1 n_2}] \quad i=1,2 \tag{6.22}$$

where

$$C_1 = \frac{n_1+n_2-2}{n_1+n_2-p-3} \tag{6.23}$$

and

$$\mathrm{Var}[D_s(\underline{x})] = C_2\left(\delta^2 + \frac{p(n_1+n_2)}{n_1 n_2}\right) \tag{6.24}$$

where

$$C_2 = \frac{(n_1+n_2-3)(n_1+n_2-2)^2}{(n_1+n_2-p-2)(n_1+n_2-p-3)(n_1+n_2-p-5)} \quad . \tag{6.25}$$

Although $D_s(\underline{x})$ is not normally distributed, for n_1 and n_2 sufficiently large; the distribution is very close to normal. Thus the expected actual error rate E may be estimated by

$$E[E(\hat{R},f)] = P_1\phi(-\gamma_1) + P_2\phi(\gamma_2) \tag{6.26}$$

where

$$\gamma_i = \frac{E[D_s(\underline{x})\,|\,\omega_i]}{[\mathrm{Var}(D_s(\underline{x}))]^{1/2}} \quad , \quad i=1,2 \quad . \tag{6.27}$$

(4) The plug-in estimate of the error rate. The plug-in error rate is obtained by using the estimated parameters, $\hat{f}$, for f.

$$E(\hat{R},\hat{f}) = P_1 \int_{\hat{R}_2} \hat{p}(\underline{x}|\omega_1)d\underline{x} + P_2 \int_{\hat{R}_1} \hat{p}(\underline{x}|\omega_2)d\underline{x} \quad . \tag{6.28}$$

For the two-class multivariate normal distribution case with equal covariance matrices, D_s is normally distributed with means and covariance equal to

$$\begin{aligned} D_s(\hat{\underline{\mu}}_1) &= [\underline{m}_1 - \tfrac{1}{2}(\underline{m}_1 + \underline{m}_2)]^t S^{-1}(\underline{m}_1 - \underline{m}_2) \\ &= \tfrac{1}{2} D^2 \quad . \end{aligned} \tag{6.29}$$

$$\begin{aligned} D_s(\hat{\underline{\mu}}_2) &= [\underline{m}_2 - \tfrac{1}{2}(\underline{m}_1 + \underline{m}_2)]^t S^{-1}(\underline{m}_1 - \underline{m}_2) \\ &= -\tfrac{1}{2}D^2 \end{aligned} \tag{6.30}$$

$$\begin{aligned} \hat{V}_D &= (\underline{m}_1 - \underline{m}_2)^t S^{-1}(\underline{m}_1 - \underline{m}_2) \\ &= (\underline{m}_1 - \underline{m}_2)^t S^{-1}(\underline{m}_1 - \underline{m}_2) \\ &= D^2 \end{aligned} \tag{6.31}$$

where

$$D^2 = (\underline{m}_1 - \underline{m}_2)^t S^{-1}(\underline{m}_1 - \underline{m}_2) \quad . \tag{6.32}$$

D^2 is the Mahalanobis distance with estimated parameters. Note that the plug-in estimate is consistent and asymptotically

efficient. However, it is not unbiased and it may be quite poor for small sample sizes.

6.3 Discussion

The reader may have noted that all the estimates in the last section, except the plug-in error rate, required that the true distributions be known, which in general are not known. The plug-in error rate is simple to calculate but is noted for being severely biased unless a large training sample set is available. For the two-class multivariate normal distribution case with equal covariance matrices, there are some approximations for the expected error rate available. One method is to approximate the distribution of D_s by a series expansion. Then replace the true parameters by the estimated values, for example, replace δ^2 by D^2. Some of these expansions are given by Okamoto [87], Sitgreaves [89] and Anderson [90,91]. However, it is not known how good these expansions are for small samples. Another method is given by Lachenbruch [92] as discussed in the last section. The approximation is achieved by replacing δ^2 with D^2 in (6.22)-(6.27).

Lachenbruch and Mickey [73] compared a number of methods for estimating the error rate, which include a plug-in estimate, Okamoto's expansion, and the leave-one-out. They found that Okamoto's expansion which replaces δ^2 by D_o^2 is the best where

$$D_o^2 = \frac{n_1+n_2-p-3}{n_1+n_2-2} D^2. \tag{6.33}$$

The second best estimate was to use D^2 for δ^2 in Okamoto's expansion. In [93], Lachenbruch claimed that his method, which is given in (6.22)-(6.27) where we replace δ^2 by D^2, is comparable to using Okamoto's expansion with D^2 replacing δ^2. Dunn [94] compares Lachenbruch's estimate [92] with the results from a Monte Carlo study and noted that it tended to be slightly conservative. In no case was the estimate 0.02 greater than the Monte Carlo estimate for the error rate.

Note that the major portion of Section 6.2 has been concerned with methods based on the normal distribution with equal covariance matrices. If the data is not normal or not close to normal, the methods do not work well and results for non-normal data do not exist. Thus, other approaches like those in Section 6.1 must be employed.

CHAPTER 7
EFFECTS OF FINITE SAMPLES ON THE PERFORMANCE OF GAUSSIAN CLASSIFIER

In Chapter 2 we designed a classifier when complete knowledge of all the relevant probabilities was available. However, in most practical situations such complete knowledge is not available, instead a set of training samples together with some vague, general knowledge about the problem is available. In this chapter, we will study how this effects the performance of the classifier for the two-class case with equal a priori probability. Assume we know that the conditional probability densities are normally distributed for both classes with equal covariance matrices. The only unknowns are the means of each class and the common covariance matrix. The classifier under consideration is the Bayes classifier (Eq. (2.30) of Chapter 2) with the sample means, $\underline{m}_i$'s, and sample covariance matrix S substituted for the true values, where

$$\underline{m}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \underline{x}_j^{(i)} \tag{7.1}$$

$$S = \frac{1}{n_1+n_2-2} \sum_{i=1}^{2} \sum_{j=1}^{n_i} (\underline{x}_j^{(i)} - \underline{m}_i)^2 \tag{7.2}$$

and $\underline{x}_j^{(i)}$ refers to the jth training sample from class ω_i. We will show that when only a finite number of training samples is available, there exists an optimal number of features.

7.1 The Effects of Unequal Training Sample Sizes

We showed in the last chapter that the expected actual error rate is given by ((6.26) of Chapter 6)

$$E[E(\hat{R},f)] = P_1\phi(-\gamma_1) + P_2\phi(\gamma_2) \tag{7.3}$$

where

$$\gamma_i = \frac{E[D_s(\underline{x})\,|\,\omega_i]}{[\mathrm{Var}(D_s(\underline{x}))]^{1/2}} \quad , \; i = 1,2 \tag{7.4}$$

$$E[D_s(\underline{x})\,|\,\omega_1] = \frac{1}{2}C_1[(-1)^{i+1}\delta^2 - \frac{p(n_2-n_1)}{n_1n_2}] \quad , \; i=1,2 \tag{7.5}$$

$$C_1 = \frac{n_1+n_2-2}{n_1+n_2-p-3} \tag{7.6}$$

$$\mathrm{Var}[D_s(\underline{x})] = C_2(\delta^2 + \frac{p(n_1+n_2)}{n_1n_2}) \tag{7.7}$$

$$C_2 = \frac{(n_1+n_2-3)(n_1+n_2-2)^2}{(n_1+n_2-p-2)(n_1+n_2-p-3)(n_1+n_2-p-5)} \quad . \tag{7.8}$$

Rearrange (7.3)-(7.8) and substitute $P_1 = P_2 = 0.5$, then we have the following. (To simplify our notation from now on we will represent the expected actual error rate $E[E(\hat{R},f)]$ by P_E.)

$$P_E = E[E(\hat{R},f)] = \frac{1}{2} P_{E1} + \frac{1}{2} P_{E2}$$

$$= \frac{1}{2} \phi[-(\alpha - \beta)] + \frac{1}{2} \phi [-(\alpha + \beta)] \tag{7.9}$$

where

$$\alpha = C_3 \frac{\delta^2}{(\delta^2 + \frac{p(n_1 + n_2)}{n_1 n_2})^{1/2}}$$

$$\beta = C_3 \frac{\frac{p(n_2 - n_1)}{n_1 n_2}}{(\delta^2 + \frac{p(n_1 + n_2)}{n_1 n_2})^{1/2}} = C_3 \frac{p(n_2 - n_1)}{(n_1 n_2(\delta^2 n_1 n_2 + p(n_1 + n_2)))^{1/2}}$$

$$C_3 = \frac{1}{2} \left(\frac{(n_1 + n_2 - p - 2)(n_1 + n_2 - p - 5)}{(n_1 + n_2 - 3)(n_1 + n_2 - p - 3)}\right)^{1/2}$$

Note that for a given $N = n_1 + n_2$, δ^2 and p, C_3 is constant. However, as $(n_2 - n_1) > 0$, is increased, α decreases and β increases. The maximum value of α and the minimum value of β occur at the same point, $n_2 - n_1 = 0$ or $n_2 = n_1$. Since $\phi(z)$ is a monotonically increasing function, then for β fixed, P_E increases as α decreases. From Figure 7.1 it is clear that as β increases, P_E increases for α fixed. Therefore, when $(n_2 - n_1) > 0$ is increased P_E increases for a given $N = n_1 + n_2$, δ^2 and p. The minimum P_E occurs when n_1 is equal to n_2. This result agrees with [95]. Also observe that when $n_2 - n_1$ increases, P_{E1} increases while P_{E2} decreases but the decreasing P_{E2} can not offset the increasing P_{E1}. This can be interpreted as follows. As $(n_2 - n_1)$ increases for a fixed $N = n_1 + n_2$, then n_2 is becoming larger and n_1 is

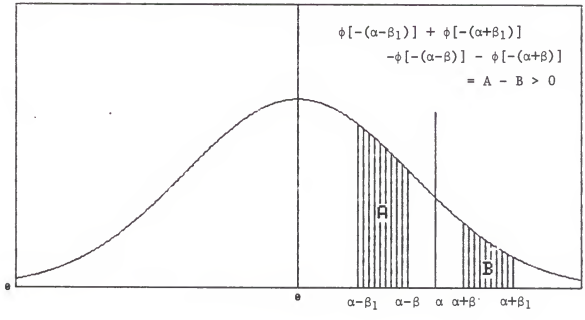


Figure 7.1 The effect of changing β.

becoming smaller. Therefore we know more about class ω_2 and less about class ω_1. Thus less errors are made when the observations come from ω_2 and more errors are made when the observations come from ω_1. However, the information gained about class ω_2 can not offset the loss in information about class ω_1.

When $N = n_1 + n_2$, $n_1 - n_2$ and δ^2 are fixed (i.e. n_1, n_2 and δ^2 are fixed), as p increases, C_3 and α decrease but β increases. Thus, the effect of an unequal training sample size is more pronounced when a large number of features is employed. This conclusion agrees with the observation in [24].

When n_1 is greater than n_2 the above results still hold, except P_{E1} and P_{E2} interchange their roles. After this section we will consider only the case where $n_1 = n_2$.

7.2 Minimum Increase in δ^2 to Avoid Peaking.

For $n_2 = n_1$, the expected actual error rate P_E in (7.9) can be simplified to

$$P_E = \phi(-\gamma) \tag{7.10}$$

where

$$\gamma = \frac{1}{2}\left(\frac{(N-2-p)(N-5-p)}{(N-3)(N-3-p)}\right)^{1/2} \frac{\delta^2}{\left(\delta^2 + \frac{4p}{N}\right)^{1/2}} \tag{7.11}$$

$$N = n_1 + n_2 = 2n$$

From (7.10) the expected actual error rate P_E is only a function of the total number of training samples N, Mahalanobis distance δ^2, and the number of features p. Note from (7.10) that the expected actual error rate decreases for a given p as either or both N and δ^2 increase but increases for a fixed N and δ^2 as p increases. Thus, if a fixed number N of training samples is available, it is interesting to know how much δ^2 should be increased to justify an additional feature. In other words, what is the least contribution to the Mahalanobis distance that an additional feature should make in order for it to be included as a new feature. Mathematically, we want to find x^2 such that $P_{E1}((\delta^2 + x^2), N, p+1) = P_{E2}(\delta^2, N, p)$. But $P_{E1} = P_{E2}$, when they have the same value of γ i.e.

$$\frac{1}{2}\left(\frac{(N-2-p)(N-5-p)}{(N-3)(N-3-p)}\right)^{1/2} \frac{\delta^2}{\left(\delta^2 + \frac{4p}{N}\right)^{1/2}} =$$

$$\frac{1}{2}\left(\frac{(N-2-p-1)(N-5-p-1)}{(N-3)(N-3-p-1)}\right)^{1/2} \frac{\delta^2 + x^2}{\left(\delta^2 + x^2 + \frac{4(p+1)}{N}\right)^{1/2}}$$

or

$$\frac{(N-2-p)(N-4-p)(N-5-p)}{(N-3-p)^2(N-6-p)} \frac{\delta^2 + x^2 + 4\left(\frac{p+1}{N}\right)}{\delta^2 + \frac{4p}{N}} = \frac{(\delta^2 + x^2)^2}{\delta^4} \tag{7.12}$$

It can be shown that a sensible solution of (7.12) is

$$x^2 = \delta^2 \left(\frac{C_4 N + \sqrt{C_4^2 N^2 + 16(\delta^2 N + 4p)(p+1)C_4}}{2(\delta^2 N + 4p)} - 1\right) \tag{7.13}$$

where

$$C_4 = \frac{(N - 2 - p)(N - 4 - p)(N - 5 - p)}{(N - 3 - p)^2(N - 6 - p)} \tag{7.14}$$

The plots of x^2 versus p for different values of δ^2 when N = 20 and 100 are shown in Figures 7.2a and 7.2b. From these figures, it is clear that when N is large, x^2 is small. On the other hand, when δ^2 or p is large, x^2 is large too. This may be explained by noting that the estimates are more reliable when more training samples are employed or less parameters are estimated. Another important feature of Figure 7.2 is that the curves have exponential character for large values of δ^2. This can be explained by rewriting (7.12) as follows.

$$\frac{(N-2-p)}{(N-3-p)}\left(1 - \frac{1}{N-3-p}\right)\left(1 + \frac{1}{N-6-p}\right)\frac{\delta^2 + x^2 + 4(\frac{p+1}{N})}{\delta^2 + \frac{4p}{N}} = \frac{(\delta^2 + x^2)^2}{\delta^4} \tag{7.15}$$

when $\delta^2 >> \frac{4p}{N}$

$$\frac{\delta^2 + x^2 + 4(\frac{p+1}{N})}{\delta^2 + \frac{4p}{N}} \approx \frac{\delta^2 + x^2}{\delta^2}$$

and

$$\left(1 - \frac{1}{N-3-p}\right)\left(1 + \frac{1}{N-6-p}\right) \approx 1$$

Hence, from (7.15) it can easily be shown that

$$x^2 = \frac{\delta^2}{N-3-p} \tag{7.16}$$

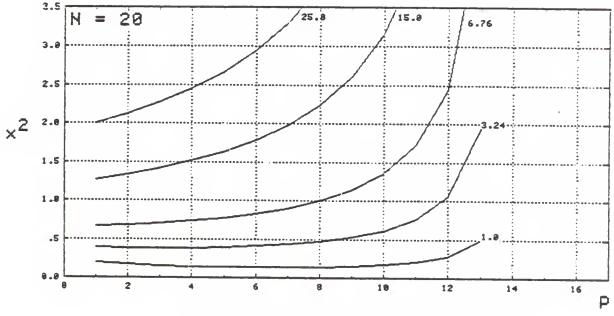


Figure 7.2a The minimal curves for N = 20.

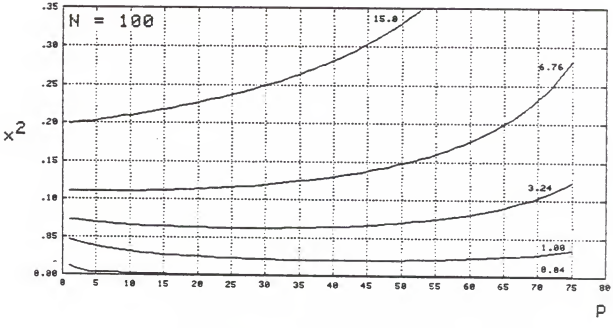


Figure 7.2b The minimal curves for N = 100.

This explains the exponential character and the properties of Fig. 7.2. All of the results in this section agree well with [24].

7.3 The Optimum Number of Features

Given N training samples (n for each class, N = 2n) and P measurements for each sample, a basic question is how many features and which features should be employed to minimize the error rate? As we noted in the previous section, the error rate depends on the total number of training samples N, the number of features p and Mahalanobis distance δ^2. In this case the number of training samples N is fixed, thus only the number of features p and the Mahalanobis distance δ^2 are variables. However, the Mahalanobis distance δ^2 depends on the p features selected. Different sets of p features may have different values of Mahalanobis distance. Recall from the last section that for a fixed number of features p, the error rate decreases as the Mahalanobis distance δ^2 increases. Thus, the problem reduces to a problem of finding p and a corresponding set of p features with the highest δ^2 among the set of all p features which minimize P_E in (7.10). Also recall that P_E is monotonically decreasing with γ, thus we can maximize γ in (7.11) or γ with the constant terms removed, then γ becomes

$$\Gamma = \left(\frac{(N-2-p)(N-5-p)}{(N-3-p)}\right)^{1/2} \frac{\delta^2}{\left(\delta^2 + \frac{4p}{N}\right)^{1/2}} \tag{7.17}$$

One way of solving this is for each p (p=1, ---, P) select the combination of p features which will yield the highest δ^2 (i.e. select the best set of p features), then obtain Γ_p from (7.17). The optimum number of features p_{opt} is the smallest p such that $\Gamma_p > \Gamma_{p+1}$ and those p_{opt} features with the highest δ^2 are the corresponding features. As was shown in the last section, this is equivalent to the problem of determining when the contribution of an additional feature to the accumulated Mahalanobis distance falls below a threshold. Note that an additional feature is always the best among the remaining features (i.e. the feature are selected from the best to worst). We now determine the optimal number of features for two special cases. These examples will clarify the points discussed earlier in this section.

Case 1. Each feature is equally good, that is, the Mahalanobis distance for any set of p features is

$$\delta_p^2 = \sum_{i=1}^{p} d^2 = pd^2 \qquad p = 1,---,P$$

where d^2 is the Mahalanobis distance for each feature. One such case is when the difference between the means is $(\underline{\mu}_1 - \underline{\mu}_2) = [d, d, ---, d]^t$ and the common covariance matrix is $\sum = I$.

After p features are selected, Γ in (7.17) becomes

$$\Gamma_p = \left(\frac{(N-2-p)(N-5-p)}{N-3-p}\right)^{1/2} \frac{pd^2}{(pd^2 + \frac{4p}{N})^{1/2}}$$

$$= \left[\left(1 + \frac{1}{N-3-p}\right)\left(1 - \frac{1}{N-4-p}\right)\right]^{1/2} \left(p(N-4-p)\right)^{1/2} \frac{d^2}{\left(d^2 + \frac{4}{N}\right)^{1/2}} \quad (7.18)$$

Note that the last term in (7.18) is independent of p and for large N the first bracket is approximately equal to 1 thus independent of p. Therefore, the p which maximizes Γ_p is the p which maximizes p(N-4-p) which is $p_{opt} = \frac{N}{2} - 2$.

Note that the result in this example is not quite the same as that in [24], where they obtained $\frac{N}{2} - 1$ (in our notation). However, the difference is small (i.e. equal to 1, to be exact). As noted in the last chapter, the Lachenbruch estimate tends to be slightly conservative.

Case 2. The contribution to the Mahalanobis distance of each feature is a fixed fraction of the contribution of the previous feature. Thus after p features are chosen, the accumulated Mahalanobis distance is

$$\delta_p^2 = d^2 + c^2 d^2 + c^4 d^2 + \text{---} + c^{2(p-1)} d^2, \quad c>0.$$

$$= d^2 \frac{1-c^{2p}}{1-c^2}$$

where d^2 is the Mahalanobis distance of the first feature. An example of this case is when the difference between the means is $(\underline{\mu}_1 - \underline{\mu}_2) = [d, cd, \text{---}, c^{p-1}d]^t$ and the common covariance matrix is $\Sigma = I$.

After p features are selected, Γ in (7.17) becomes

$$\Gamma_p = \left(\frac{(N-2-p)(N-5-p)}{(N-3-p)}\right)^{1/2} \frac{d^2 \frac{1-c^{2p}}{1-c^2}}{\left(d^2 \frac{1-c^{2p}}{1-c^2} + \frac{4p}{N}\right)^{1/2}} \tag{7.19}$$

The plots of Γ versus p when N=30 and d=1 for various values of c are shown in Fig. 7.3. It can be seen that increasing c leads to a better classification and a larger optimal number of features. One important characteristic of Fig. 7.3 is that when c is small, the curves reach a peak at a very small value of p. But when $c > 1$, i.e., the successive features are rated better or contribute more significantly to correct classification, causing the peak to occur at a very large value of p. This might lead one to think that the peaking phenomena can be avoided by selecting features from worst to best. Fig. 7.5 shows the plots of Γ versus p when the features are selected from best to worst as well as from worst to best. It is clear that even though the peak of the worst to best ordering occurs later, the performance of best to worst ordering is always better (i.e., higher Γ), except when all the features are employed; in which case the performance is the same for both cases.

Figure 7.4 shows the plots of Γ versus p where $d^2 = 1.0$ and $c = 0.90$ for various values of N. Note that increasing N improves the performance and increases the value of the optimal number of features required for classification. Table 7.1 shows the optimal number of features for various values of c, N and d^2. Note that for a small value of c, increasing N does not significantly increase the optimal number of features, but when N is large changing d^2 does change the optimal number

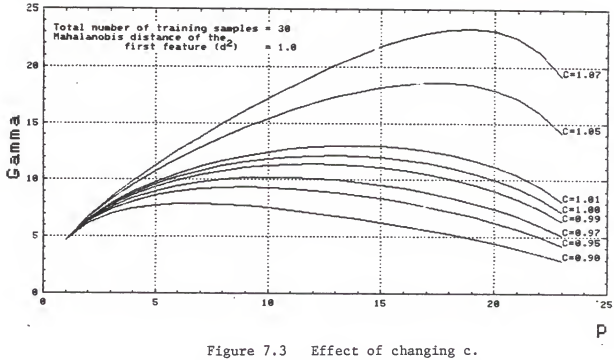


Figure 7.3 Effect of changing c.

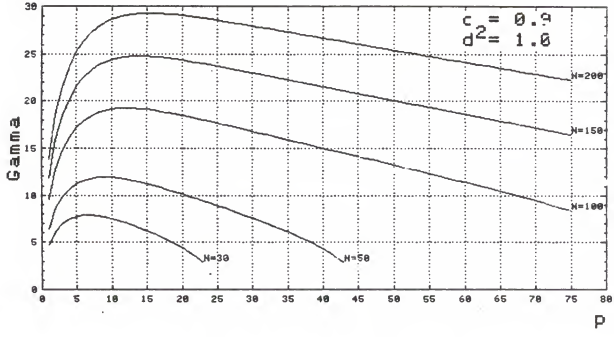


Figure 7.4 Effect of changing N.

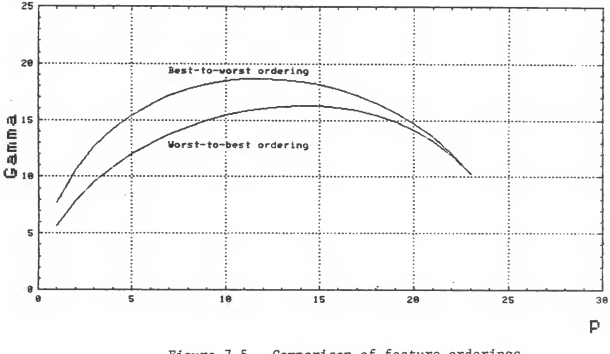


Figure 7.5 Comparison of feature orderings.

Table 7.1 The optimal number of features for Case 2.

C	N	Mahalanobis distance 0.04	0.50	1.0	10.0	25.0
	20	2	3	3	4	4
	50	3	5	5	6	6
0.80	100	4	6	7	8	8
	150	4	7	8	9	9
	200	4	7	8	10	10
	20	3	4	4	5	5
	50	4	6	7	8	8
0.85	100	5	8	9	10	10
	150	5	9	10	11	12
	200	6	10	11	12	13
	20	4	5	5	5	5
	50	6	8	9	10	10
0.90	100	7	11	12	14	14
	150	8	13	14	16	16
	200	8	14	15	17	17
	20	5	6	6	6	6
	50	10	13	13	14	14
0.95	100	13	18	20	21	21
	150	15	22	23	25	25
	200	16	25	26	28	28
	20	7	7	7	8	8
	50	19	20	20	20	20
0.99	100	33	37	38	38	38
	150	44	51	52	52	53
	200	53	62	63	64	64
	20	8	8	8	8	8
	50	23	23	23	23	23
1.00	100	48	48	48	48	48
	150	73	73	73	73	73
	200	98	98	98	98	98
	20	8	8	8	8	8
	50	27	26	26	26	26
1.01	100	64	61	61	61	61
	150	106	103	102	102	102
	200	150	148	148	148	148
	20	12	11	11	11	11
	50	40	40	40	40	40
1.10	100	90	90	90	90	90
	150	140	140	140	140	140
	200	190	190	190	190	190

of features. When c is large ($\approx$1 and larger) the optimal number of features does not depend on d^2, but depends on the number of training samples N. This result agrees well with [24].

7.4 The Optimal Number of Features for Several Covariance Matrix Structures

In the previous section, we considered two types of functions relating the Mahalanobis distance δ^2 and the number of features p regardless of the structure of the covariance matrix and the mean vectors. In this section we consider three types of Toeplitz matrices. For simplicity, let us assume that the difference between the means for each feature is the same (and they also have the same sign), i.e. $\underline{\mu}_2 - \underline{\mu}_1 = (d, d, d, ---, d)^t$. Thus, the Mahalanobis distance is the sum of all elements of $\sum^{-1}$ times d^2. It will be shown that even in this simple case, it is difficult or impossible to order the features in the optimum way (i.e. from best to worst).

Example 1. Equal correlation covariance matrix. This form of covariance matrix is given by

$$\sum = \begin{bmatrix} 1 & \rho & \rho & \cdot & \cdot & \cdot & \rho \\ \rho & 1 & \rho & & & & \cdot \\ \rho & \rho & 1 & & & & \cdot \\ \cdot & & & \cdot & & & \cdot \\ \cdot & & & & \cdot & & \cdot \\ \cdot & & & & & 1 & \rho \\ \rho & \cdot & \cdot & \cdot & \cdot & \rho & 1 \end{bmatrix}, \quad \text{where } -\frac{1}{p-1} < \rho < 1.$$

It can be shown that the inverse of this matrix is given by

$$\sum^{-1} = \begin{bmatrix} x & y & y & \cdot & \cdot & \cdot & y \\ y & x & y & & & & \cdot \\ \cdot & & & & & & \cdot \\ \cdot & & & & & & \cdot \\ \cdot & & & & & & \cdot \\ \cdot & & & & & x & y \\ y & \cdot & \cdot & \cdot & \cdot & y & x \end{bmatrix}$$

where

$$x = \frac{1 + (p-2)\rho}{(1-\rho)(1 + \rho(p-1))}$$

$$y = \frac{-\rho}{(1-\rho)(1 + \rho(p-1))}$$

Thus we obtain

$$\delta_p^2 = \frac{pd^2}{\rho(p-1) + 1} \tag{7.20}$$

Note that in this case all of the features are equally good. Any set of p features will give the same Mahalanobis distance δ_p^2 which is given by (7.20). However, for different values of p, each additional feature contributes to the Mahalanobis distance a different amount. This can be seen from $\delta_{p+1}^2 - \delta_p^2$ which is given by

$$\delta_{p+1}^2 - \delta_p^2 = \frac{d^2(1 - \rho)}{(\rho p+1)(\rho p+1 - \rho)} \tag{7.21}$$

For $\rho > 0$, (7.21) shows that $\delta^2_{p+1} - \delta^2_p$ is monotonically decreasing as p increases. This implies that the increase in the accumulated Mahalanobis distance becomes less with each additional feature. Thus, in this case, we can select the features arbitrarily but still, in effect, have the features in the best-to-worst ordering. When $\rho < 0$ the situation becomes reversed. The increase in the accumulated Mahalanobis distance becomes greater with each additional feature. Therefore, it is impossible to order the features in a best-to-worst ordering. This also implies that for $\rho < 0$, peaking is not really a problem since it occurs at p greater than $\frac{N}{2} - 2$. Thus, we will determine the optimal number of features for the case of $0 < \rho < 1$ only.

For $0 < \rho < 1$, δ^2_p decreases as ρ increases for a fixed p. So we would expect peaking to occur at a smaller p when ρ is larger. When ρ is small, δ^2_p is almost proportional to p, i.e. $\delta^2_{p+1} - \delta^2_p \approx$ constant, thus peaking should occur at $p \approx \frac{N}{2} - 2$. Then for a larger ρ, the peaking should occur at $p < \frac{N}{2} - 2$. The optimal number of features for various values of ρ, d^2 and N are shown in Table 7.2. This table shows that when ρ is large, an additional feature is almost useless. When ρ is small, the optimal number of features is almost independent of d^2. For large d, Γ in (7.17) can be rewritten similarly to (7.18) thus reducing the problem to a problem of maximizing the expression

$$(N-4-p)\ \frac{d^2 p}{\rho(p-1) + 1}$$

which gives

Table 7.2 The optimal number of features for Example 1.

d^2	N	ρ = 0.001	0.01	0.1	0.3	0.5	0.8
	20	8	7	5	2	1	1
	100	47	37	13	4	2	1
0.1	200	93	66	18	6	3	1
	500	221	128	29	10	5	2
	1000	408	203	41	13	7	3
	20	8	7	5	3	2	1
	100	47	39	19	9	5	2
1.0	200	93	71	30	13	8	3
	500	223	142	50	22	13	5
	1000	412	227	74	31	18	8
	20	8	8	6	4	3	2
	100	47	40	21	12	8	4
10.0	200	94	72	33	18	12	6
	500	223	143	57	30	20	10
	1000	413	230	85	44	28	14

$$p_{opt} = \frac{(\rho-1) + \sqrt{(\rho-1)^2 + \rho(N-4)(1-\rho)}}{\rho}$$

Example 2. The covariance matrix in this example has the following form.

$$\Sigma = \begin{bmatrix} 1 & \rho & \rho^2 & \cdot & \cdot & \cdot & \rho^{p-1} \\ \rho & 1 & \rho & \cdot & \cdot & \cdot & \cdot \\ \rho^2 & \rho & 1 & \rho & \cdot & \cdot & \cdot \\ \cdot & & & \cdot & & & \\ \cdot & & & & \cdot & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot \\ \rho^{p-1} & \cdot & \cdot & \cdot & \cdot & \rho & 1 \end{bmatrix} \tag{7.22}$$

Thus the correlation between features x_i and x_j is equal to $\rho^{|i-j|}$. For simplicity we will consider only the case where $0 < \rho < 1$. Note that in this case, when each feature is used alone, all features are equally good. However, different sets of p features may not give the same value for the Mahalanobis distance. For example, the Mahalanobis distance of any two features x_i and x_j is monotonically increasing with $|i-j|$, since the correlation between two features is lower if they are farther apart. Thus, the best combination of two features is the first and the last one, and the worst combination is any two consecutive features.

Jain and Waller [24] and El-Sheikh and Wacker [25] have considered this problem. They find the optimal number of features p_{opt} to be the order of the covariance matrix, p, in (7.22) that minimizes the expected

probability of error. In effect, they select the features in a "natural" order. It is shown in [25] that an additional feature always increases the accumulated Mahalanobis distance by $\delta^2_{p+1} - \delta^2_p = \frac{1-\rho}{1+\rho} d^2$ for $p \geq 1$. Thus the p_{opt} is approximately (but always less than) $\frac{N}{2} - 2$ (since the first feature contributes $d^2 > \frac{1-\rho}{1+\rho} d^2$). However, from the above discussion, it is clear that values given by [24,25] are higher than the true ones. In this case, p_{opt} also depends on the total number of features available, P, in addition to d^2, ρ, and N. The larger value of P, the better the performance and the larger p_{opt} will be for a fixed ρ, d^2 and N. One of the best ways to select features in this case is the first one first, the last one second, the middle one third, etc., choosing each successive measurement (feature) to be "maximally" distant from all previously selected measurements. The plots of Γ versus p for the "optimal" and "natural" orderings when $d^2 = 1$, $\rho = 0.5$, $N = 20$ and $P = 13$ are shown in Fig. 7.6. We can see that the "optimal" value always gives a better performance except, of course, when p=1 or 13, i.e., one or all features are used. The effect of changing P for a given d^2, ρ, and N is shown in Fig. 7.7. The optimum number of features for various values of N, P, ρ, and d^2 is shown in Table 7.3.

Example 3. First order Makov covariance matrix. The form of the covariance matrix is given by

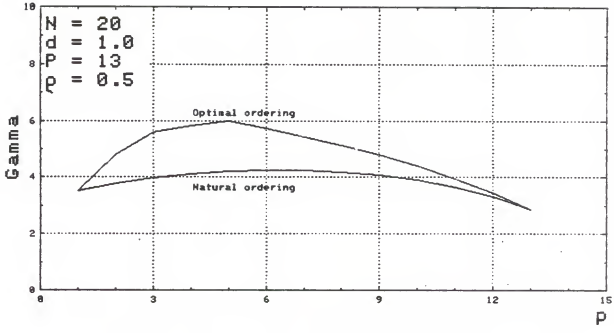


Figure 7.6 Comparison of feature orderings for Example 2.

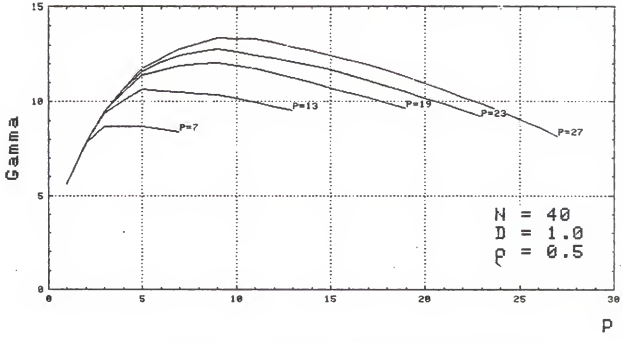


Figure 7.7 Effect of changing P for Example 2.

Table 7.3 The optimal number of features for Example 2.

P	σ^2_d	N	ρ = 0.1	ρ = 0.5	ρ = 0.9
		20	6	3	2
	0.10	40	13	8	2
		80	30	16	3
		20	6	4	2
N/2	1.00	40	15	9	3
		80	33	17	5
		20	6	4	2
	5.00	40	15	9	3
		80	33	17	5
		20	5	2	1
	0.10	40	10	4	2
		80	20	9	2
		20	5	3	2
N/4	1.00	40	10	5	2
		80	20	12	3
		20	5	3	2
	5.00	40	10	6	2
		80	20	12	3

$$\Sigma = \begin{bmatrix} 1 & \rho & & & & \\ \rho & 1 & \rho & & 0 & \\ & \rho & 1 & \rho & & \\ & & & \ddots & & \\ & & & & \ddots & \rho \\ & & 0 & & \rho & 1 \end{bmatrix}$$

where $\rho < \frac{1}{2}$ for large p. For this type of covariance matrix, correlation exists only between adjacent features. Again, the discussion in Example 2 applies here. Jain and Waller [24] and El-Sheikh and Wacker [25] have also addressed this problem. Jain and Waller [24] show that $\delta^2_{p+1} - \delta^2_p \approx d^2/(1+2\rho)$. Thus, p_{opt} is again approximately equal to $\frac{N}{2} - 2$.

Note that in this case all the odd features are uncorrelated with each other. All the even features are also uncorrelated with each other. Thus it seems reasonable that all the odd features should be chosen first since the number of the odd features is always greater than or equal to the total number of even features. In this case, p_{opt} also depends on the total number of features available, P. The comparison of the two ordering methods is shown in Fig. 7.8 for N = 20, d^2 = 1, ρ = 0.3 and P = 13. Fig. 7.9 shows the effect of changing P for a fixed N, d^2, and ρ.

When P $\geq$ N - 4, there are at least $\frac{N}{2} - 2$ uncorrelated features. In either the even or odd feature set, each feature contributes equally to the accumulated Mahalanobis distance and is equal to d^2. Thus in this case $p_{opt} = \frac{N}{2} - 2$. Observe that when either all the odd or even features are chosen, for large ρ, an additional feature from another set contributes only a small amount to the accumulated Mahalanobis

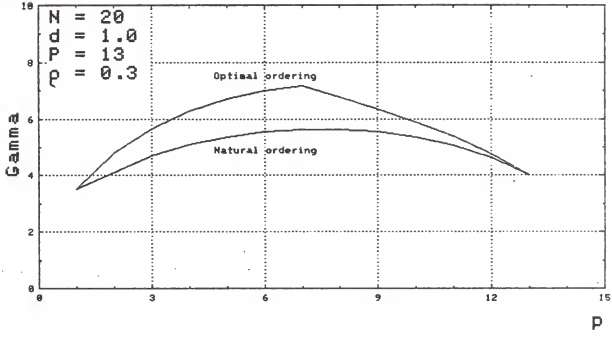


Figure 7.8 Comparison of feature orderings for Example 3.

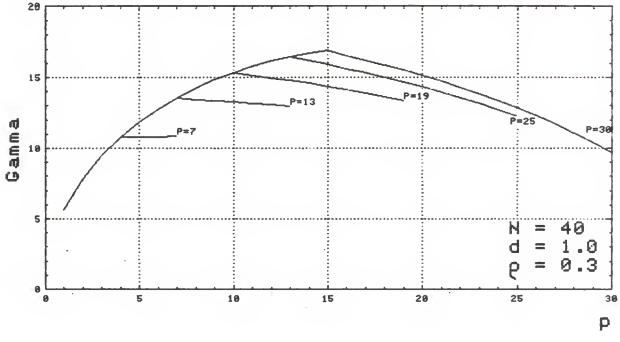


Figure 7.9 Effect of changing P for Example 3.

distance. Thus for large ρ and $P < N-4$, P_{opt} is approximately $(N+1)/2$ when N is odd and $N/2$ when N is even. The optimum number of features for various values of d^2, N, ρ, and P are shown in Table 7.4.

7.5 Remarks

1. It must be emphasized that the word "feature" here is synonymous with "measurement" or "observation." It is not a weighted combination of measurements. This implies that we must use feature selection to reduce the number of features (measurements) for avoiding the dimensionality problem. The feature extraction algorithm is used to reduce the number of featues, simplifying the problem.

2. The discussion in this section is valid only when the difference between the means for the two classes is equal in both sign and magnitude in all directions and the correlation coefficients (ρ's) are positive. If any of these conditions is violated, the discussion above is invalid and the situation becomes more complicated. For example, in Example 1, if the difference between the means for each feature is equal in magnitude but of different sign, then all features are not equally good. As another example, if the correlation coefficients (ρ's) are negative, the feature with the largest magnitude of $|\rho|$ may be the best feature.

3. From the examples above it is clear that in general there is no systematic way of selecting features. The best way is to try

Table 7.4 The optimal number of features for Example 3.

			ρ		
P	d^2	N	0.06	0.1	0.3
		20	6	5	5
	0.10	40	16	13	10
		80	34	30	20
		20	7	6	5
N/2	1.00	40	16	15	10
		80	35	33	20
		20	7	7	5
	5.00	40	17	15	10
		80	35	33	20
		20	5	5	3
	0.10	40	10	10	5
		80	20	20	10
		20	5	5	3
N/4	1.00	40	10	10	5
		80	20	20	20
		20	5	5	3
	5.00	40	10	10	10
		80	20	20	20

all the combinations. However, the amount of calculation required is enormous even for a modest number of features. Note that the classification rule (2.30) of Chapter 2 can be written as

$$x \in \omega_1 \text{ if } D = \underline{x}^t \sum^{-1} (\underline{\mu}_1 - \underline{\mu}_2) - \frac{1}{2} (\underline{\mu}_1 - \underline{\mu}_2)^t \sum^{-1} (\underline{\mu}_1 - \underline{\mu}_2) > 0.$$

$x \in \omega_2$ otherwise.

where $\underline{x}$ is the measurement vector and $\underline{x}^t = (x_1, x_2, ---, x_p)$. This rule can be interpreted as follows. If the weighted sum of measurements x_i is greater than a threshold, we decide $x \in \omega_1$, otherwise $x \in \omega_2$. Thus a significant measurement should have a large weight. Then we can order the measurements by the value of the weighting vector, $\sum^{-1} (\underline{\mu}_1 - \underline{\mu}_2)$. A measurement with the highest weighting factor should then be selected first.

7.6 Conclusion

The peaking phenomenon for equiprobable multivariate Gaussian data with equal covariance matrix has been investigated. We have shown that peaking can be avoided if an additional feature increases the value of Γ in (7.17). This condition has been shown to be equivalent to determining when the increase in the Mahalanobis distance is smaller

than a threshold given by (7.13). The condition for the optimal number of features for three types of covariance matrices has been studied. We have emphasized the importance of feature ordering an suggested that features can be selected in the same order as the order of the weighting factor value, $\sum^{-1}(\underline{\mu}_1 - \underline{\mu}_2)$. For a fixed total number of training samples, we suggest using an equal training sample size for each class.

CHAPTER 8
PATTERN RECOGNITION OF EEG

The application of pattern recognition to ERP data is discussed in this chapter. The goal of our experiments is to assess the feasibility of visual evoked response to determine the objective validity of texual materials as read by human subjects. The experiment described here is one of many experiments conducted at the Mind-Machine Interaction Research Center, Department of Electrical Engineering, University of Florida which have been reported in several papers and reports [12,96-102]. We briefly describe the experimental design and data collection procedure in Sections 8.1 and 8.2 respectively. Then in Section 8.3 we discuss the data analysis technique and compare the various methods. The results are discussed in Section 8.4.

8.1 Experimental Design

The rationale for this experimental design was given by Childers et al. [96]. In this experiment, we recorded cortical potentials while subjects decided if the statements presented were true or false. The subjects learned the true value of these statements during a familiarization phase preceding data collection. The purpose of the experiment was to determine if the pattern of difference between ERPs for true and false statements could be observed for information acquired recently for both responding and non-responding subjects.

One day prior to collections of ERPs, the subjects learned a set of 18 statements (statements number 1-9 and 19-27 in Table 8.1) about fictitious people and their occupations, e.g., "Scott is a lawyer". The next day, they were shown statements consisting of correct or "true" pairings of names and occupations, i.e. those 18 statements learned in the previous day or with "false" or mismatched names and occupations (e.g., "Scott is a singer"), i.e. statement numbers 10-18 and 28-36, while the EEG was recorded. Statements were randomly presented. In each session, each statement was presented twice. There were a total of 72 trials; 36 trials corresponded to true statements and 36 trials to false statements. Four sessions of data were collected for each subject. The subjects were asked to respond truthfully in two sessions and not to respond in the other two sessions.

8.2 Data Collection Procedure

We monitored and digitized data from five scalp electrode locations: Fz, C3, C4, Cz and Pz in the 10/20 EEG system, from one timing channel, and from an EOG channel. Each EEG location was a monopolar derivation to the linked mastoids. The data were digitized at the rate of 125 samples/sec/channel (or 8 msec sampling interval). The amplifiers had a bandwidth from 1 to 50 Hz, with 50% attenuation at 1 and 50 Hz [103]. Presentation of stimuli and collection of data was controlled by a NOVA 4 computer. Details of the software can be found in [104].

TABLE 8.1 Statements Employed in the Experiment

1.	SCOTT	IS A	LAWYER.	19.	GWEN	IS A	WRITER.
2.	PAUL	IS A	DOCTOR.	20.	FLO	IS A	DENTIST.
3.	JIM	IS A	STUDENT.	21.	JOAN	IS A	MANAGER.
4.	DAN	IS A	BROKER.	22.	SUE	IS A	TEACHER.
5.	MARTIN	IS A	CLERK.	23.	DIANE	IS A	CHEMIST.
6.	STEVEN	IS A	SINGER.	24.	BARBARA	IS A	SENATOR.
7.	ROBERT	IS A	FARMER.	25.	MARSHA	IS A	JUDGE.
8.	WALTER	IS A	MECHANIC.	26.	ELLEN	IS A	BANKER.
9.	LARRY	IS A	SCIENTIST.	27.	JUDY	IS A	DANCER.
10.	SCOTT	IS A	SINGER.	28.	GWEN	IS A	BANKER.
11.	PAUL	IS A	CHEMIST.	29.	FLO	IS A	STUDENT.
12.	JIM	IS A	FARMER.	30.	JOAN	IS A	SCIENTIST.
13.	DAN	IS A	DENTIST.	31.	SUE	IS A	LAWYER.
14.	MARTIN	IS A	JUDGE.	32.	DIANE	IS A	BROKER.
15.	STEVEN	IS A	DOCTOR.	33.	BARBARA	IS A	MANAGER.
16.	ROBERT	IS A	DANCER.	34.	MARSHA	IS A	CLERK.
17.	WALTER	IS A	SENATOR.	35.	ELLEN	IS A	TEACHER.
18.	LARRY	IS A	MECHANIC.	36.	JUDY	IS A	WRITER.

Prior to the experiment, the subject was given the appropriate instructions relevant to the experiment while the electrodes were being placed on the scalp. A 50 microvolt, 10 Hz calibrate signal was connected to the input of all the EEG amplifiers, prior to the subject being connected to the amplifier, to calibrate all recording channels. The subject was then seated in the ventilated Faraday shielded screen room and the electrodes connected to the amplifiers.

The first ER trial was a photic evoked response using a Grass Instrument xenon photo stimulator. The subject was then presented the sentences.

The sentences were presented via an HP 2468A graphic terminal which was connected to the TV monitor viewed by the subject. The TV monitor was located approximately 32" from the subject's eyes. The contrast and brightness were adjusted to each subject's specification for viewing comfort. The sentence was presented in three segments to minimize eye movement, e.g.,

SCOTT

IS A

LAWYER

The sequencing of the presentation of the sentence segments, with respect to data acquisition, is shown in Table 8.2. The total data digitized for each sentence was 4.1 sec or 512 samples for each channel.

The subject was instructed to respond (response session) by actuating a switch within 1 sec after the last sentence segment appears.

TABLE 8.2 Timing Diagram

MSEC	EXAMPLE	EVENT
		Fixation Box Appears
0		Begin Data Collection
400	SCOTT	Subject Presented
800		Subject Erased
1200	IS A	Verb Presented
1600		Verb Erased
2000	LAWYER	Object Presented
2400		Object Erased
2000-4048		Response Executed
4048		End Data Collection
4048		Message (If Any) Presented
4448		Message Erased
4448-7500		Write Data To Disk And Start Next Presentation

The timing of the segments and the subject's response were collected as one digitized data channel. Fz, Cz, timing and EOG were also monitored, via a Grass Instruments Polygraph, on-line during the experiments.

Eight subjects have participated in this experiment. Each subject attended four sessions of 72 sentences and responded truthfully in two sessions, and made no response in the other two sessions. Thus we have two sets of data for each subject, one is the "response" data and the other is "no response" data. Each set consists of two classes of data; data corresponding to the true sentences and data corresponding to false sentences. For the response data set these two classes are called true response (TR) and false response (FR). Similarly for the no-response data, one class is called true no-response (TN) and the other is called false no-response (FN).

8.3 Data Analysis Techniques

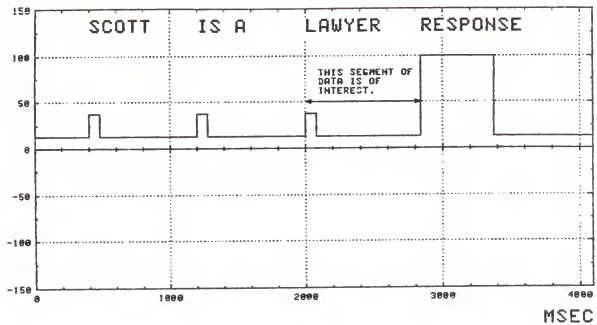
Each data set described in the last section was analyzed separately. The data in each set was partitioned into two subsets, one was used to design a classifier, the other was used for testing, i.e. the holdout estimate described in Section 6.1 was used for estimating the error rate. Since each data set contained two sessions of 72 sentences, we partitioned the data by using one session for design and the other session for testing. Thus we had 36 sentences for each class (i.e. 36 training samples for each class) for designing and testing. This number is generally considered to be small especially when compared

with the number of features, 512 for each training sample. However, we were only interested in discriminating the true ERPs from the false ERPs. Thus, we need only look at the data after the last stimulus and before the response is made (for response session), see Figure 8.1. Consequently, the number of features could potentially be reduced to less than 262, which is still very large. Note that the length of data (i.e. the number of features) for each response session is not the same for all sessions, since each subject takes a differing amount of time to answer each statement. Thus we need to align the data. In this study, three data alignment techniques are considered.

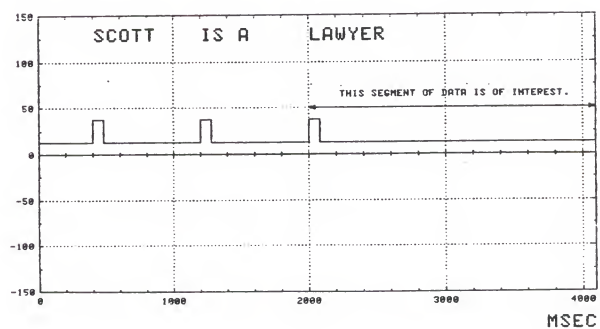
8.3.1 Data Alignment Techniques

Three data alignment techniques, stimulus locked (S-locked), response locked (R-locked) and linear time warping are considered. Each technique is based on different assumptions. The segment of the ERP between the last stimulus and the response can be divided into three portions (see Figure 8.2). The first portion is of duration τ_1 which represents the time delay between the presentation of the last image and the initiation of the subject's thinking process. The second segment, τ_2 is the time required to make a decision. The last segment, τ_3, is the time required to make a response. Ideally, we would like to use the data from the second segment only. However, we do not know when this segment occurs. Hopefully, the data alignment techniques properly align the data, so that the second segment can be identified.

Figure 8.1 The segment of data use for analysis (a) for response session (b) for non-response session.


SCOTT
IS A
LAWYER
RESPONSE
THIS SEGMENT OF DATA IS OF INTEREST.
150
100
50
0
-50
-100
-150
0
1000
2000
3000
4000
MSEC

(a)


SCOTT
IS A
LAWYER
THIS SEGMENT OF DATA IS OF INTEREST.
150
100
50
0
-50
-100
-150
0
1000
2000
3000
4000
MSEC

(b)

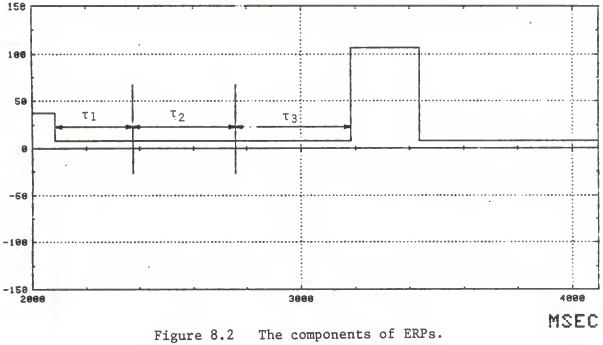


Figure 8.2 The components of ERPs.

(1) Stimulus locked (S-locked). Under this method the data are locked to the last stimulus presentation (2000 msec). This technique is based on the assumption that τ_1 (the time delay between the last stimulus presentation and the start of the thinking process) is almost constant. Then, when the data are locked to the stimulus, the second portion, τ_2 (the thinking and the decision making process) will align.

(2) Response locked (R-locked). Under this method the data are locked to the response point (arbitrarily plotted as 2800 msec). This technique is based on the assumption that τ_3 (the time delay required to make a response) is almost constant.

(3) Time warping (T-locked). A portion of the signal from the last stimulus presentation (object of the sentence) to the response point is either linearly stretched or compressed to make all trials of equal length (120 samples or 960 msec). Thus, for example, if the last stimulus for a particular trial occurs at 2000 msec, the data are either linearly compressed or stretched so that the response becomes located at 2960 msec. This technique is based on the assumption that the ratio of $\tau_1 : \tau_2 : \tau_3$ is constant, i.e. when the subject responded late to any trial, this means that for that particular trial he was slow in thinking about his answer, slow in making a decision, and also slow in making the response.

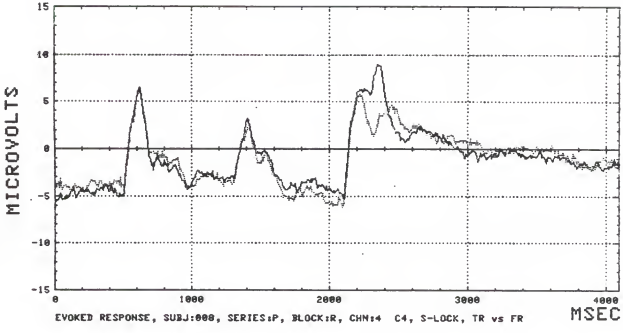
The plots of the average ERPs over all 8 subjects when the data are synchronized by S-locked, R-locked and linear time warping for channels 4, 6 and timing for response sessions are shown in Figures

8.3, 8.4 and 8.5 respectively. The data for S-locked, channels 4, 6 and timing for no-response sessions are shown in Figure 8.6.

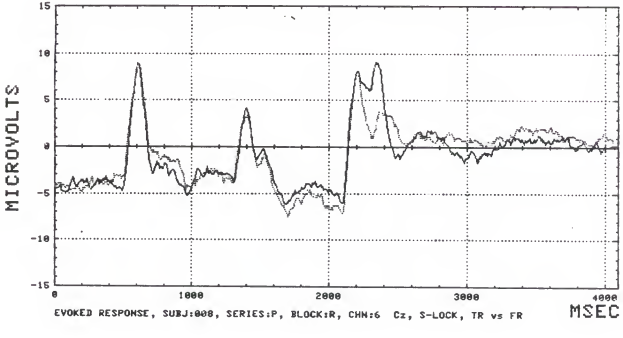
8.3.2 Features Selection and Extraction of ERPs

At this point, we have some one hundred features or so for each training sample. The discussion in the last subsection indicates that the segment of data we really want is the second portion, which lies somewhere between the last stimulus and the response. Thus, we must take a segment of data from that region for analysis. The averages in Figures 8.3-8.6 can help us locate this position of the data. From these figures, we see that a large difference between the two classes occurs in the region 2200-2400 msec for S-locked data, 2500-2700 for R-locked data and 2400-2600 msec for the time warped data. This gives 25 features for each trial (training sample).

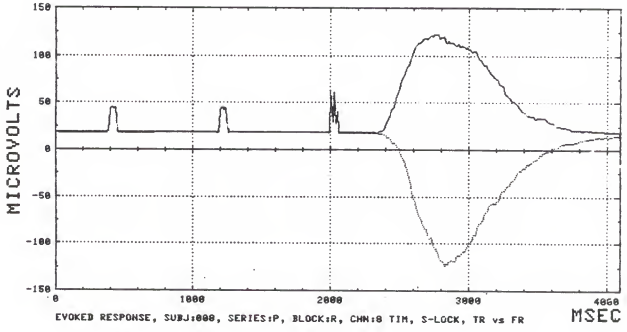
This number of features can be reduced to the optimal number by using a feature selection technique. Since only 36 training samples are available for each class (with 25 features), it is not feasible to use the nonparametric approach to estimate the conditional probability densities. The parametric approach must be employed and the form of the conditional probability density must be assumed. In this case we will assume that the data are normally distributed with equal covariance matrices for both classes. The feature selection procedure described in Section 7.4 of Chapter 7 can be used, i.e. the features are ordered by the value of $\sum^{-1}(\underline{\mu}_1 - \underline{\mu}_2)$ where the covariance matrix $\sum$ and means $\underline{\mu}_1$ and $\underline{\mu}_2$ are approximated by the sample covariance matrix S ((3.10) of


15
10
5
0
-5
-10
-15
MICROVOLTS
0
1000
2000
3000
4000
MSEC
EVOKED RESPONSE, SUBJ:008, SERIES:P, BLOCK:R, CHN:4 C4, S-LOCK, TR vs FR

(a)

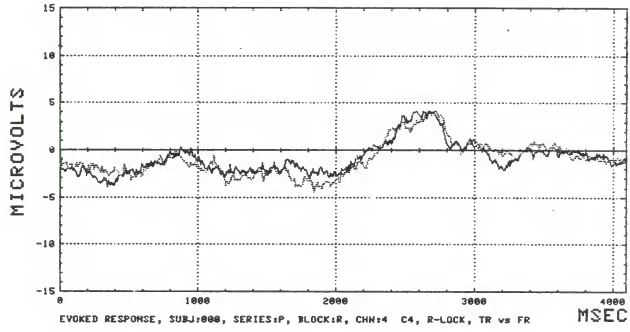

15
10
5
0
-5
-10
-15
MICROVOLTS
0
1000
2000
3000
4000
MSEC
EVOKED RESPONSE, SUBJ:008, SERIES:P, BLOCK:R, CHN:6 Cz, S-LOCK, TR vs FR

(b)

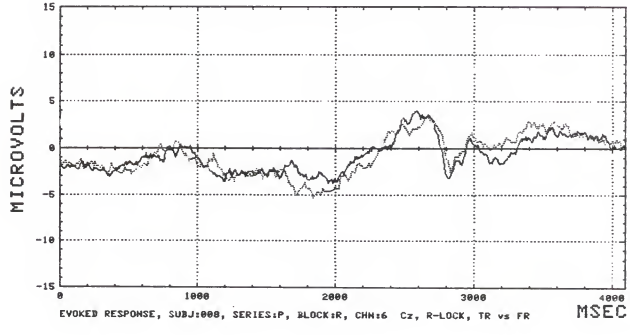


(c)

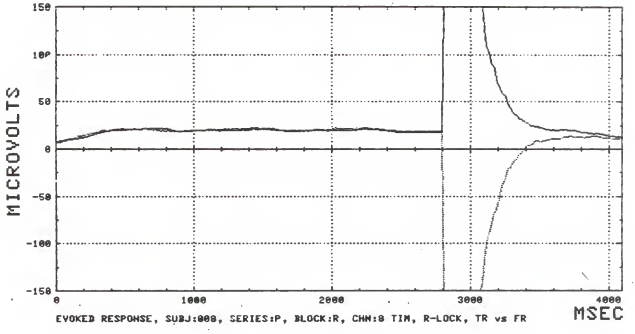
Figure 8.3 Average of stimulus-locked data. The solid line is the average of true response (TR), while the dotted line is the average of false response (FR) for (a) channel 4; (b) channel 6; (c) timing.


MICROVOLTS
15
10
5
0
-5
-10
-15
0
1000
2000
3000
4000
MSEC
EVOKED RESPONSE, SUBJ:000, SERIES:P, BLOCK:R, CHN:4 C4, R-LOCK, TR vs FR

(a)

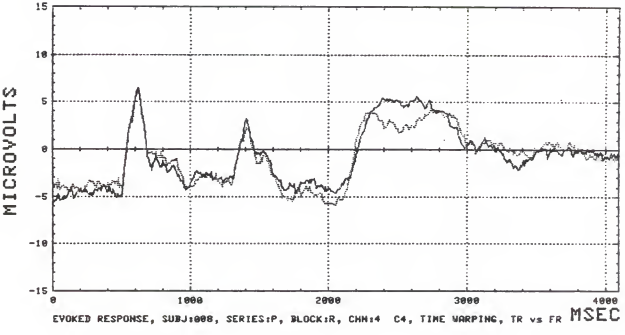

MICROVOLTS
15
10
5
0
-5
-10
-15
0
1000
2000
3000
4000
MSEC
EVOKED RESPONSE, SUBJ:000, SERIES:P, BLOCK:R, CHN:6 Cz, R-LOCK, TR vs FR

(b)

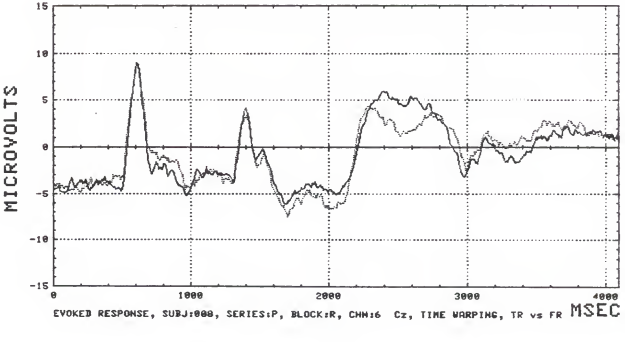


(c)

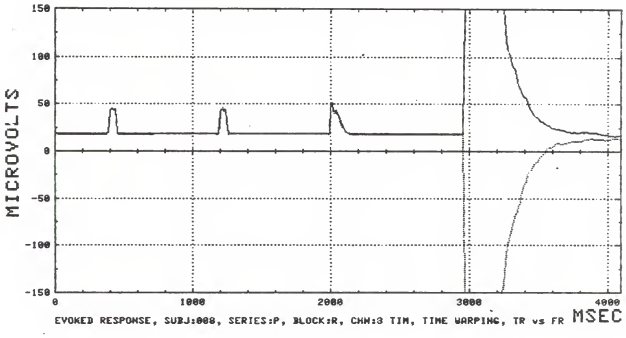
Figure 8.4 Average of response-locked data. The data have all been shifted arbitrarily so that all responses are aligned at 2800 ms. The solid line is the average of true response (TR), while the dotted line is the average of false response (FR) for (a) channel 4; (b) channel 6; (c) timing.


MICROVOLTS
15
10
5
0
-5
-10
-15
0
1000
2000
3000
4000
MSEC
EVOKED RESPONSE, SUBJ:008, SERIES:P, BLOCK:R, CHN:4 C4, TIME WARPING, TR vs FR

(a)

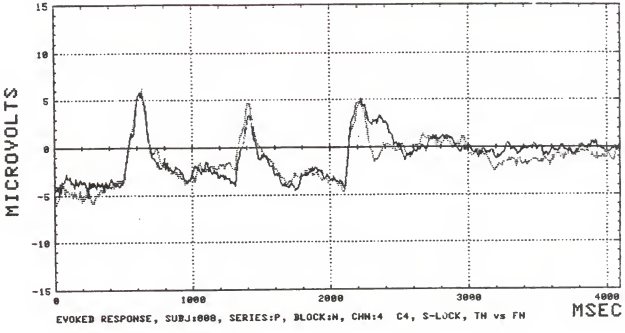

MICROVOLTS
15
10
5
0
-5
-10
-15
0
1000
2000
3000
4000
MSEC
EVOKED RESPONSE, SUBJ:008, SERIES:P, BLOCK:R, CHN:6 Cz, TIME WARPING, TR vs FR

(b)

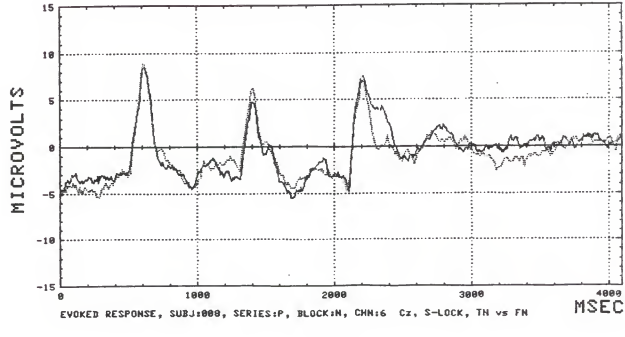


(c)

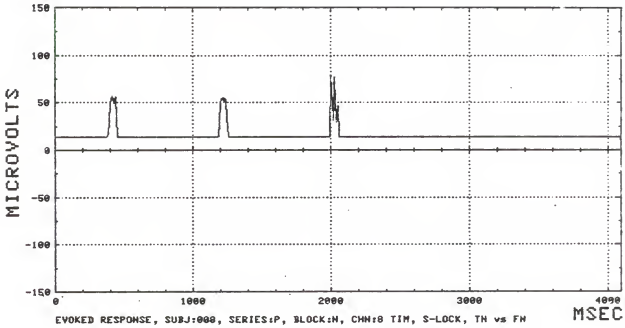
Figure 8.5 Average of time warping data. The data have all been either stretched or compressed so that all responses are aligned at 2960 ms. The solid line is the average of true responses (TR), while the dotted line is the average of false responses (FR) for (a) channel 4; (b) channel 6; (c) timing.


MICROVOLTS
15
10
5
0
-5
-10
-15
0
1000
2000
3000
4000
MSEC
EVOKED RESPONSE, SUBJ:000, SERIES:P, BLOCK:N, CHN:4 C4, S-LOCK, TH vs FH

(a)


MICROVOLTS
15
10
5
0
-5
-10
-15
0
1000
2000
3000
4000
MSEC
EVOKED RESPONSE, SUBJ:000, SERIES:P, BLOCK:N, CHN:6 Cz, S-LOCK, TH vs FH

(b)



(c)

Figure 8.6 Average of stimulus-locked data for no response sessions. The solid line is the average of TN, while the dotted line is the average of FN for (a) channel 4; (b) channel 6; (c) timing.

Chapter 3) and sample means $\underline{m}_i$ ((3.8) of Chapter 3). The optimal number features and the corresponding features are determined by (7.17) of Chapter 7, i.e., finding a p such that

$$\Gamma_p = \left(\frac{(N-2-p)(N-5-p)}{(N-3-p)}\right)^{1/2} \frac{\delta^2}{\left(\delta^2+\frac{4p}{N}\right)^{1/2}}$$

is maximized, where

$N = 72$

δ^2 is approximated by $D^2 = (\underline{m}_1 - \underline{m}_2)^t S^{-1} (\underline{m}_1 - \underline{m}_2)$.

The optimal number of features for this data set is approximately 3 or 4.

To simplify the design of the classifier, we reduce the number of features to two by employing the Foley-Sammon method [59] described in Section 5.2 of Chapter 5. Consequently, each training sample (or each trial) is represented by two features. Then these two features are plotted in two dimensional space. A plot of all the training samples is called a scatter plot. Then a classifer, a line which separates the two classes is drawn. This classifier is used to classify the testing data set which is not employed in the design stage to determine the holdout error rate. A sample figure of merit table, which shows the figure of merit for each feature, for the optimal number of features for Channel 4 and 6 is shown in Table 8.3. In this table, the label "optimal number of features" indicates the method described above was used while label "all features" indicates a method that used all the features, i.e. 25

Table 8.3 Figure of Merit

Fisher Ratio (ALL FEATURES ARE USED) Class: TR vs FR

Subj:PJO, Ser:P, Sess:1, S-LOCK (2200 - 2400), 36 Samp/Cl.

		NO OF FEAT	1ST VECTOR	2ND VECTOR
CHN:4	C4	25	8.86961	4.77280
CHN:6	Cz	25	8.52324	4.80850

Fisher Ratio (OPTIMAL NUMBER OF FEATURES) Class: TR vs FR

Subj:PJO, Ser:P, Sess:1, S-LOCK (2200 - 2400), 36 Samp/Cl.

		NO. OF FEAT	1ST VECTOR	2ND VECTOR
CHN:4	C4	1	1.92600	1.92600
CHN:6	Cz	1	3.77417	0.00000

without determining the optimal number of features (i.e. apply Foley-Sammon method directly to reduce the number of features from 25 to 2). Figure 8.7 summarizes the procedure for both methods. A sample scatter plot is shown in Figure 8.8. The error rate of both design and test sets for S-locked, R-locked and time warping for response sessions are shown in Table 8.4, 8.5 and 8.6 respectively. The error rate for no response sessions for S-locked is shown in Table 8.7.

8.4 Results

From Tables 8.4-8.7, the results can be summarized as follows.

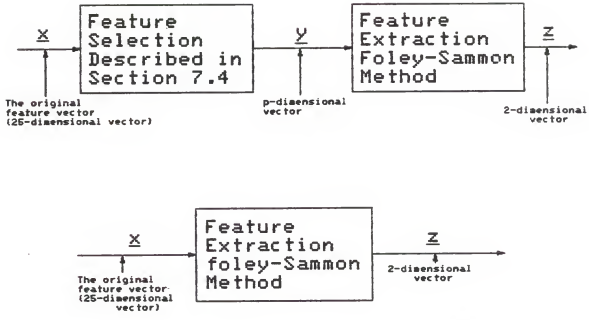
(1) The error rate, when the optimum number of features is used, is on the average better than when all features are used. When all features are used, there is a large difference between the error rate of the design and test sets. This result verifies the derivation in Chapter 7 and confirms the existence of an optimal number of features.

(2) For the sessions when the subject responds with a finger lift, the S-locked data is the best.

(3) On the average, the response sessions have lower error rates than the non-response sessions. (A response session is when the subject indicates his response with a finger lift.) This result agrees with [102] which indicated that the response session shows a larger difference than the non-response session.

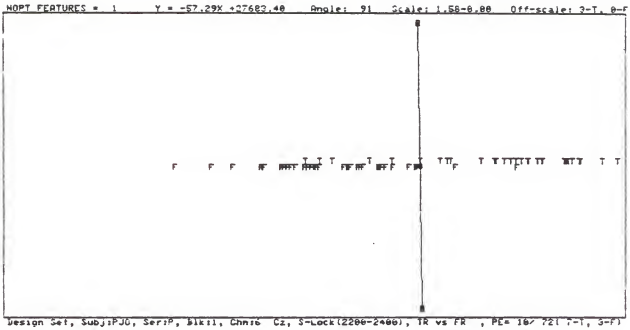
(4) The error rate varies form subject to subject. The best subjects are PJO for response sessions and JWB for non-response sessions.

(5) Channel 4 (C4) is slightly better than channel 6 (Cz).

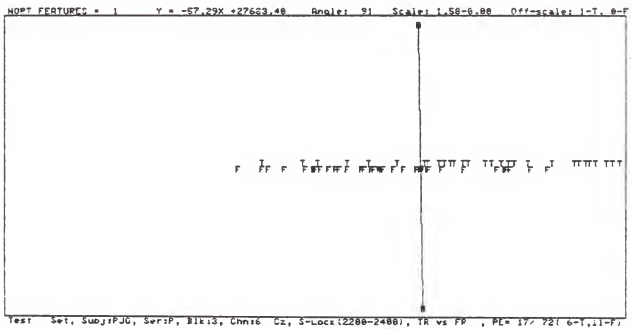


(b)

Figure 8.7 A summary of feature reduction techniques, (a) when optimal number of features is used; (b) when all features are used.



(a)



(b)

Figure 8.8 Scatter plot; (a) for design set (b) for test set

TABLE 8.4 Number of Errors for the True Response (TR) vs False Response (FR) S-Locked (2200-2400), 72 Trials

SUBJECT		CHN 4 OPT	C4 ALL	CHN 6 OPT	CZ ALL
PJO	D	14	3	10	3
	T	17	22	17	22
VFH	D	20	8	15	11
	T	27	29	34	30
SMR	D	27	18	28	7
	T	30	40	30	43
DAT	D	34	14	29	11
	T	26	25	28	30
RHM	D	15	8	18	11
	T	26	24	29	26
JWB	D	21	14	26	18
	T	31	29	25	29
MPG	D	20	14	15	15
	T	26	29	36	38
JPM	D	23	19	17	12
	T	41	37	35	39
AVERAGE NUMBER OF ERRORS	D	21.75	12.25	19.75	11.00
	T	28.00	29.38	29.25	32.13
ERROR RATE IN %	D	30.21	17.01	27.43	15.28
	T	38.89	40.80	40.63	44.62

TABLE 8.5 Number of Errors for True Response (TR) vs. False Response (FR) R-locked (2500-2700), 72 Trials

SUBJECT		CHN 4 OPT	C4 ALL	CHN 6 OPT	CZ ALL
PJO	D	22	13	23	9
	T	26	33	41	42
VFH	D	32	17	25	17
	T	33	40	40	40
SMR	D	22	12	23	15
	T	38	34	39	40
DAT	D	25	9	17	11
	T	38	34	32	34
RHM	D	35	14	17	8
	T	37	31	38	36
JWB	D	23	14	23	14
	T	36	44	47	45
MPG	D	23	14	28	19
	T	34	31	34	26
JPM	D	26	12	24	11
	T	36	42	32	38
AVERAGE NUMBER OF ERRORS	D	26	13.13	22.50	13.00
	T	34.75	36.13	37.88	37.63
ERROR RATE IN %	D	36.11	18.23	31.25	18.06
	T	48.26	50.17	52.60	52.26

TABLE 8.6 Number of Errors for True Response (TR) vs. False Response (FR) Time-Warping (2400-2600), 72 Trials

SUBJECT		CHN 4 OPT	C4 ALL	CHN 6 OPT	Cz ALL
PJO	D	15	10	28	11
	T	32	30	33	29
VHF	D	22	10	19	14
	T	37	36	39	33
SMR	D	32	15	30	16
	T	32	29	34	29
DAT	D	28	18	29	18
	T	28	35	34	32
RHM	D	25	15	22	16
	T	32	41	40	37
JWB	D	26	18	31	18
	T	43	38	33	36
MPG	D	24	18	27	11
	T	37	38	38	36
JPM	D	22	12	21	14
	T	32	42	35	33
AVERAGE NUMBER OF ERRORS	D	24.25	14.50	25.08	14.75
	T	34.13	36.13	35.75	33.13
ERROR RATE IN %	D	33.68	20.14	35.93	20.49
	T	47.40	50.17	49.65	46.01

TABLE 8.7 Number of Error for True No-Response (TN) vs. False No-Response (FN) S-Locked (2200-2400), 72 Trials

SUBJECT		CHN4 OPT	C4 ALL	CHN 6 OPT	CZ ALL
PJO	D	30	13	36	19
	T	30	35	38	34
VFH	D	27	18	24	14
	T	31	35	29	34
SMR	D	23	15	32	14
	T	38	35	39	28
DAT	D	29	20	30	17
	T	36	40	35	35
RHM	D	30	13	24	15
	T	32	33	24	33
JWB	D	19	12	28	12
	T	22	23	29	22
MPG	D	21	13	19	17
	T	39	36	34	27
JPM	D	29	22	20	17
	T	29	34	39	35
AVERAGE NUMBER OF ERRORS	D	26.00	15.75	26.63	15.63
	T	32.13.	33.88	33.38	31.00
ERROR RATE IN %	D	36.11	21.88	36.98	21.70
	T	44.62	47.05	46.35	43.06

CHAPTER 9
CONCLUDING REMARKS AND FUTURE WORK

The major results of this research are summarized as follows. 1) The derivation of a relationship between the optimal number of features and the training sample size for the two-class Gaussian data case with equal covariance matrices where means are unknown as well as the covariance matrices. 2) We suggested a feature selection technique and emphasized the importance of feature ordering. 3) We studied the effect of unequal training sample size. 4) We applied the classification techniques to ERP data and compared different data alignment techniques.

Some interesting future research in this area would be to 1) study the relationship between the optimal number of features and training sample size for other cases of Gaussian data with unequal covariance matrices, 2) use the leave-one-out or Jack Knife method to estimate the error rate, 3) find some new data alignment technique, for example, non-linear time warping, and 4) use some technique to normalize the data to decrease the variance.

REFERENCES

[1] E. Donchin, Data analysis techniques in average evoked potential research, in Average Evoked Potentials: Methods, Results, and Evaluations. Edited by E. Donchin and D. B. Lindsley, Washington, D.C.: Govern. Printing Office, pp. 199-236, 1969.

[2] G. Pfurtscheller and R. Cooper, Selective averaging of the intracerebral click evoked responses in man: An improved method of measuring latencies and amplitudes, Electroenceph. Clin. Neurophysiol., vol. 38, pp. 187-190, 1975.

[3] J. I. Aunon, C. D. McGillem, and D. G. Childers, Signal processing in evoked potential research: Averaging and modeling, Critical Reviews in Bioengineering, vol. 5, pp. 323-367, July 1981.

[4] E. Donchin and R. I. Herning, A simulation study of the efficacy of stepwise discriminant analysis in the detection and comparison of event related potentials, Electroenceph. Clin. Neurophysiol., vol. 38, pp. 51-68, 1975.

[5] K. C. Squires and E. Donchin, Beyond averaging: The use of discriminant functions to recognize event-related potentials elicited by single auditory stimuli, Electroenceph. Clin. Neurophysiol., vol. 41, pp. 449-459, 1976.

[6] R. L. Horst and E. Donchin, Beyond averaging II. Single trial classification of exogenous event-related potentials using stepwise discriminant analysis, Electroenceph. Clin. Neurophysiol., vol. 48, pp. 113-126, 1980.

[7] J. I. Aunon and C. D. McGillem, Techniques for processing single evoked potentials, Proc. San Diego Biomed. Symp., pp. 211-218, 1975.

[8] C. D. McGillem and J. I. Aunon, Measurements of signal components in single visually evoked brain potentials, IEEE Trans. Biomed. Engr., vol. BME-24, pp. 232-241, May 1977.

[9] R. W. Sencaj, J. I. Aunon, and C. D. McGillem, Discrimination among visual stimuli by classification of their single evoked potentials, Med. Biol. Eng. Comp. vol. 17, pp. 391-396, 1979.

[10] C. D. McGillem, J. I. Aunon, and D. G. Childers, Signal processing in evoked potential research: Applications of filtering and pattern recognition, Critical Reviews in Bioengineering, vol. 6, pp. 225-265, October 1981.

[11] J. J. Vidal, Real-time detection of brain events in EEG, Proc. IEEE, vol. 65, pp. 633-641, May 1977.

[12] D. G. Childers, P. A. Bloom, A. A. Arroyo, S. E. Roucos, I. S. Fischler, T. Achariyapaopan, and N. W. Perry, Jr., Classification of cortical responses using features from single EEG records, IEEE Trans. Biomed. Engr., vol. BME-29, pp. 423-438, June 1982.

[13] K. Fukunaga, Introduction to Statistical Pattern Recognition. New York: Academic Press, 1972.

[14] R. O. Duda and P. E. Hart, Pattern Classification and Scene Analysis. New York: Wiley, 1973.

[15] E. A. Patrick, Fundamentals of Pattern Recognition. Englewood Cliffs, New Jersey: Prentice-Hall, 1972.

[16] L. N. Kanal, Patterns in pattern recognition, IEEE Trans. Info. Theory, vol. IT-20, pp. 697-722, Nov. 1974.

[17] D. C. Allais, The problem of too many measurements in pattern recognition, IEEE Int. Conv. Rec., Part 7, pp. 124-130, 1966.

[18] G. F. Hughes, On the mean accuracy of statistical pattern recognizers, IEEE Trans. Inform. Theory, vol. IT-14, pp. 55-63, Jan. 1968.

[19] G. V. Trunk, A problem of dimensionality: A simple example, IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-1, pp. 306-307, July 1979.

[20] B. Chandrasekaran and A. K. Jain, Quantization complexity and independent measurements, IEEE Trans. Comput. vol. C-23, pp. 102-106, Jan. 1974.

[21] L. Kanal and B. Chandrasekaran, On dimensionality and sample size in statistical pattern classification, Pattern Recognition, vol. 3, pp. 225-234, 1971.

[22] J. W. Van Ness and C. Simpson, On the effects of dimension in discriminant analysis, Technometrics, vol. 18, no. 2, pp. 175-187, May 1976.

[23] J. W. Van Ness, On the effects of dimension in discriminant analysis for unequal covariance populations, Technometrics, vol. 21, no. 1, pp. 119-127, Feb. 1979.

[24] A. K. Jain and W. G. Waller, On the optimal number of features in the classification of multivariate Gaussian data, Pattern Recognition, vol. 10, pp. 365-374, 1978.

[25] T. S. El-Sheikh and A. G. Wacker, Effect of dimensionality and estimation on the performance of Gaussian classifier, Pattern Recognition, vol. 12, pp. 115-126, 1980.

[26] S. Raudys and V. Pikelis, On dimensionality, sample size, classification error, and complexity of classification algorithm in pattern recognition, IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-2, pp. 242-252, May 1980.

[27] S. E. Roucos and D. G. Childers, On dimensionality and learning set size in feature extraction, Proc. 1980 Internat'l Conf. on Cybernetics and Society. pp. 26-31, Oct. 1980.

[28] S. E. Roucos, On Small Sample Performance of Pattern Recognition Machine, Ph.D. Dissertation, University of Florida, 1980.

[29] H. L. Van Trees, Detection, Estimation, and Modulation Theory - Part I, New York: Wiley, 1968.

[30] C. W. Helstrom, Statistical Theory of Signal Detection, Oxford: Pergamon Press, 1975.

[31] A. Wald, Statistical Decision Functions, New York: Wiley, 1950.

[32] D. Blackwell and M. A. Girshick, Theory of Games and Statistical Decision, New York: Wiley, 1954.

[33] T. W. Anderson, An Introduction to Multivariate Statistical Analysis, New York: Wiley, 1958.

[34] H. Chernoff and L. E. Moses, Elementary Decision Theory, New York: Wiley, 1959.

[35] T. S. Ferguson, Mathematical Statistics: A Decision Theoretic Approach. New York: Academic Press, 1967.

[36] G. S. Sebestyen, Decision-Making Processes in Pattern Recognition. New York: MacMillan, 1962.

[37] N. J. Nilsson, Learning Machines. New York: McGraw-Hill, 1965.

[38] K. S. Fu, Sequential Methods in Pattern Recognition and Machine Learning. New York: Academic Press, 1968.

[39] J. M. Mendel and K. S. Fu, eds. Adaptive, Learning and Pattern Recognition Systems. New York: Academic Press, 1970.

[40] K. S. Fu, Syntactic Methods in Pattern Recognition. New York: Academic Press, 1974.

[41] S. Watanabe, ed. Methodologies of Pattern Recognition. New York: Academic Press, 1969.

[42] W. S. Meisel, Computer-Oriented Approaches to Pattern Recognition. New York: Academic Press, 1972.

[43] H. C. Andrews, Introduction to Mathematical Techniques in Pattern Recognition. New York: Wiley, 1972.

[44] C. H. Chen, Statistical Pattern Recognition. Rochelle Park, New Jersey: Hayden Book Co., 1973.

[45] J. T. Tou and R. C. Gonzalez, Pattern Recognition Principles, Reading, Massachusetts: Addison-Wesley, 1974.

[46] T. Y. Young and T. W. Calvert, Classification, Estimation and Pattern Recognition. New York: American Elsevier, 1974.

[47] P. A. Devijver and J. Kittler, Pattern Recognition: A Statistical Approach. London: Prentice-Hall, 1982.

[48] A. M. Mood, F. A. Graybill and D. C. Boes, Introduction to the Theory of Statistics, Third edition, New York: McGraw-Hill, 1974.

[49] J. D. Gibbons, Nonparametric Statistical Inference, New York: McGraw-Hill, 1971.

[50] E. Parzen, On estimation of a probability density function and mode, Ann. Math. Stat., vol. 32, pp. 1065-1076, Sept. 1962.

[51] F. Rosenblatt, The perceptron-a perceiving and recognizing automation, Report 85-460-1, Cornell Aeronautical Laboratory, Ithaca, N.Y., Jan. 1957.

[52] A. B. J. Novikoff, On convergence proofs for perceptrons, Symposium on Mathematical Theory of Automata, Polytechnic Institute of Brooklyn, vol. 12, pp. 615-622, 1963.

[53] C. R. Rao and S. K. Mitra, Generalized Inverse of Matrices and its Applications. New York: Wiley, 1971.

[54] J. S. Koford and G. F. Groner, The use of an adaptive threshold element to design a linear optimal pattern classifier, IEEE Trans. Info. Theory, vol. IT-12, pp. 42-50, Jan. 1966.

[55] J. D. Patterson and B. F. Womack, An adaptive pattern classification system, IEEE Trans. Sys. Sci. Cyb., vol. SSC-2, pp. 62-67, Aug. 1966.

[56] Y. C. Ho and R. L. Kashyap, A class of iterative procedures for linear inequalities, J. SIAM Control, vol. 4, pp. 112-115, 1966.

[57] R. A. Fisher, The use of multiple measurements in taxonomic problems, Ann. Eugenics, vol. 7, Part II, pp. 179-188, 1936.

[58] F. R. Gantmacher, The Theory of Matrices, vol. 1, New York: Chelsea Publishing company, 1959.

[59] D. H. Foley and J. W. Sammon, Jr., An optimal set of discriminant vectors, IEEE Trans. Comput., vol. C-24, pp. 281-289, March 1975.

[60] A. N. Mucciardi and E. E. Gose, A comparison of seven techniques for choosing subsets of pattern recognition properties, IEEE Trans. Comput., vol. C-20, pp. 1023-1031, Sept. 1971.

[61] S. Kullback, Information Theory and Statistics. New York: Wiley, 1959.

[62] P. A. Devijver, On a new class of bounds on Bayes risk in multihypothesis pattern recognition, IEEE Trans. Comput., vol. C-23, pp. 70-80, 1974.

[63] T. R. Vilmansen, Feature evaluation with measures of probability dependence, IEEE Trans. Comput., vol. C-22, pp. 381-388, 1973.

[64] L. Kanal, Patterns in pattern recognition: 1968-1974, IEEE Trans. Inform. Theory, vol. IT-20, pp. 697-722, Nov. 1974.

[65] W. B. Davenport and W. L. Root, Introduction of Random Signals and Noise. New York: McGraw-Hill, 1958.

[66] Y. T. Chien and K. S. Fu, On the generalized Karhunen-Loève expansion, IEEE Trans. Inform. Theory (Corrsp.), vol. IT-13, pp. 518-520, July 1967.

[67] J. T. Tou and R. P. Heydorn, Some approaches to optimum feature extraction, in Computers and Information Sciences - II. Edited by J. T. Tou, New York: Academic Press, 1967.

[68] S. Watanabe, P. F. Lambert, C. A. Kulikowski, J. L. Buxton and R. Walker, Evaluation and selection of variables in pattern recognition, Computers and Information Sciences-II, Edited by J. T. Tou, New York: Academic Press, pp. 91-122, 1967.

[69] K. Fukunaga and W. L. G. Koontz, Application of the Karhunen-Loève expansion to feature selection and ordering, IEEE Trans. Comput., vol. C-19, pp. 311-318, April 1970.

[70] S. E. Roucos and D. G. Childers, Feature extraction for small design sets: Two new algorithms, Pre-published, 1981.

[71] G. T. Toussaint, Bibliography on estimation of misclassification, IEEE Trans. Inform. Theory, vol. IT-20, pp. 472-479, July 1974.

[72] M. Hill, Allocation rules and their error rates, J. Roy. Stat. Soc., Ser. B, vol. 28, pp. 1-31, 1966.

[73] P. A. Lachenbruch and R. M. Mickey, Estimation of error rates in discriminant analysis, Technometrics, vol. 10, pp. 1-11, 1968.

[74] D. H. Foley, Considerations of sample and feature size, IEEE Trans. Inform. Theory, vol. IT-18, pp. 618-626, Sept. 1972.

[75] K. Fukunaga and D. L. Kessel, Estimation of classification error, IEEE Trans. Comput., vol. 20, pp. 1521-1527, Dec. 1971.

[76] S. John, The distribution of Walds classification statistic when the dispersion matrix is known, Sankhyā, vol. 21, pp. 371-376, 1960.

[77] S. John, On some classification statistics, Sankhyā, vol. 22, pp. 309-317, 1960.

[78] S. John, Errors in discrimination, Ann. Math. Statist., vol. 32, pp. 1125-1144, 1961.

[79] R. Sitgreaves, Some results on the distribution of the W-classification statistic, in Studies in Item Analysis and Prediction. Edited by H. Solomon, Stanford, Calif.: Stanford University Press, pp. 241-251, 1961.

[80] A. Wald, On a statistical problem arising in the classification of an individual into one of two groups, Ann. Math. Statist., vol. 15, pp. 145-162, 1944.

[81] T. W. Anderson, Classification by multivariate analysis, Psychometrika, vol. 16, pp. 31-50, 1951.

[82] H. L. Harter, On the distribution of Wald's classification statistics, Ann. Math. Statist., vol. 22, pp. 58-67, 1951.

[83] R. Sitgreaves, On the distribution of two random matrices used in classification procedures, Ann. Math. Statist., vol. 23, pp. 263-270, 1952.

[84] D. Teichroew and R. Sitgreaves, Computation of an empirical sampling distribution for the W-classification statistic, in Studies in Item Analysis and Prediction, Edited by H. Solomon, Stanford, Calif.: Stanford University Press, pp. 252-275, 1961.

[85] A. H. Bowker, A representation of Hotelling's T^2 and Anderson's classification statistic W in terms of simple statistics, in Studies in Item Analysis and Prediction, Edited by H. Solomon, Stanford, Calif.: Stanford University Press, pp. 285-292, 1961.

[86] A. H. Bowker and R. Sitgreaves, An asymptotic expansion for the distribution function of the W-classification statistic, in Studies in Item Analysis and Prediction, Edited by H. Solomon, Stanford, Calif.: Stanford University Press, pp. 293-310, 1961.

[87] M. Okamoto, An asymptotic expansion for the distribution of the linear discriminant function, Ann Math. Statist., vol. 34, pp. 1286-1301, 1963. Correction: Ann Math. Statist., vol. 39, pp. 1358-1359, 1968.

[88] D. G. Kabe, Some results on the distribution of two random matrices used in classification procedures, Ann. Math. Statist., vol. 34, pp. 181-185, 1963.

[89] R. Sitgreaves, Some operating characteristics of linear discriminant functions, in Discriminant Analysis and Applications. Edited by T. Calcoullos, New York: Academic Press, pp. 365-374, 1973.

[90] T. W. Anderson, Asymptotic evaluation of the probabilities of misclassification by linear discriminant functions, in Discriminant Analysis and Applications. Edited by T. Calcoullos, New York: Academic Press, pp. 17-35, 1973.

[91] T. W. Anderson, An asymptotic expansion of the distribution of the 'studentized' classification statistic W, Ann. Stat., vol. 1, pp. 964-972, 1973.

[92] P. A. Lachenbruch, On expected probabilities of misclassification in discriminant analysis, necessary sample size, and a relation with the multiple correlation coefficient, Biometric, vol. 24, pp. 823-834, 1968.

[93] P. A. Lachenbruch, Discriminant Analysis. New York: Hafner Press, 1975.

[94] O. J. Dunn, Some expected values for probabilities of correct classification in discriminant analysis, Technometrics, vol. 13, pp. 345-353, May 1971.

[95] B. Chandrasekaran and A. K. Jain, Independence, measurement complexity and classification performance, IEEE Trans. Syst., Man Cybern., vol. SMC-5, pp. 240-244, 1975.

[96] D. G. Childers, I. S. Fischler and N. W. Perry, Jr., Visual Evoked Response and Cognition, First Annual Report, Contract 78-F-295000, Department of Electrical Engineering, University of Florida, September 1979, Revised November 1979.

[97] D. G. Childers, I. S. Fischler and N. W. Perry, Jr., Visual Evoked Response and Cognition, Second Annual Report, Contract 78-F-295000, Department of Electrical Engineering, University of Florida, September 1980.

[98] D. G. Childers, I. S. Fischler and N. W. Perry, Jr., Visual Evoked Response and Cognition, Third Annual Report, Contract 78-F-295000, Department of Electrical Engineering, University of Florida, October 1981.

[99] D. G. Childers, I. S. Fischler and N. W. Perry, Jr., Visual Evoked Response and Cognition, Fourth Annual Report, Contract 78-F-295000, Department of Electrical Engineering, University of Florida, October, 1982.

[100] I. S. Fischler, P. A. Bloom, D. G. Childers, S. E. Roucos, and N. W. Perry, Jr., Brain potentials related to stages of sentence verification, Psychophysiology, vol. 20, pp. 400-409, 1983.

[101] I. S. Fischler, P. A. Bloom, D. G. Childers, S. E. Roucos, and N. W. Perry, Jr., Late negativity and semantic incongruity: brain potentials related to stages of sentence verification, Submitted for publication, 1983.

[102] I. S. Fischler, D. G. Childers, T. Achariyapaopan, and N. W. Perry, Jr., Brain potentials during sentence verification: II. Automatic aspects of comprehension, submitted for publication, 1983.

[103] D. G. Childers, Evoked responses: Electrogenesis, models methodology, and wavefront reconstruction and tracking analysis, Proc. IEEE, vol. 65, pp. 611-626, May 1977.

[104] A. A. Arroyo, Implementation, Collection, and Processing of Visual Evoked Responses (CPVER) for Cognitive Studies, Ph.D. Dissertation, University Florida, 1981.

BIOGRAPHICAL SKETCH

Teera Achariyapaopan was born in Thailand on September 10, 1955. He received the Bachelor of Engineering degree from Chulalongkorn University, Bangkok, Thailand, in 1976, and the Master of Science degree in electrical engineering from Youngstown State University, Youngstown, OH, in 1979.

Since September, 1979, he has been with the Department of Electrical Engineering at the University of Florida.

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Donald D. Childers, Chairman
Professor of Electrical Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Thomas E. Bullock
Professor of Electrical Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Leon W. Couch
Associate Professor of Electrical Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Nathan W. Perry, Jr.
Professor of Clinical Psychology

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Martin A. Uman
Professor of Electrical Engineering

This dissertation was submitted to the Graduate Faculty of the College of Engineering and to the Graduate Council, and was accepted as partial fulfillment of the requirements for the degree of Doctor of Philosophy.

December 1983

Dean, College of Engineering

Dean, Graduate School